RWTH AACHEN UNIVERSITY

GazeTouch:
Using Gaze Tracking to Select Indirect Touch Targets

Master's Thesis at the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Andrii Matviienko

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr. Johannes Schöning

Registration date: Sep 12th, 2014
Submission date: Dec 5th, 2014

I hereby declare that I have created this work completely on my own and used no other sources or tools than the ones listed, and that I have marked any citations accordingly.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, December 2014
Andrii Matviienko

Contents

List of Figures

List of Tables

Abstract

For most of the people in different research areas the workspace consists of horizontal and vertical surface. Usually horizontal surface is used for placing the input devices, documents or everyday objects, and vertical is used for the presentation of the output information. In particular, touch screens surfaces transform the workspace into one interactive environment, where both surfaces are used for data input and output for the variety of tasks and situations. In the recent years a lot of research has been done in the area of interactive touch screen surfaces, in order to improve an interaction efficiency and usage convenience.

We present a combination of horizontal and vertical touch surfaces in the form of a single digital workspace . Using a traditional direct touch technique in such a setup can become exhausting and lead to less accurate interaction over a long period of usage, what is known as Gorilla Arm effect. In order to overcome these problems and ease an interaction, we present two gaze-based interaction techniques, eliminating the need of using direct interaction with a vertical screen. With gaze-based interaction techniques the direct touch is only needed for a horizontal screen.

The newly developed gaze-based techniques are called ITSS and ITOS. Indirect Touch Screen Selection (ITSS) allows users to select the screen they intend to interact by simply looking at it and *absolutely* maps the touch input from the horizontal to the vertical screen. Indirect Touch Object Selection (ITOS) highlights objects the user is looking at and uses *relative* direct touch mapping. After all, we compare these two eye-gaze based techniques with a traditional direct touch on the system, which combines one vertical and one horizontal touch screen, where both screens are used as input and output surfaces.

Überblick

Für die meisten Menschen besteht der Arbeitsbereich, ungeachtet der verschiedenen Forschungsbereiche, in denen sie tätig sind, aus einer horizontalen und einer vertikalen Oberfläche. Gewöhnlich wird die horizontale Fläche zum Anordnen der Eingabegeräte, Dokumente oder Alltagsgegenstände genutzt, während die vertikale Fläche für die Präsentation der Ausgabeinformationen verwendet wird. Touchscreen-Oberflächen verwandeln den Arbeitsplatz in eine interaktive Umgebung, in welcher beide Oberflächen für Dateneingabe und –Ausgabe, sowie für die Bewältigung vielfältiger Aufgaben und Situationen verwendet werden. In den letzten Jahren wurde viel Forschung auf dem Gebiet der interaktiven Touchscreen-Oberflächen betrieben, um Effizienz und Nutzungskomfort zu verbessern.

Wir präsentieren eine Kombination von horizontalen und vertikalen Touch-Oberflächen im Form eines einzigen digitalen Arbeitsplatzes. Die Nutzung einer Technik mit traditionellem direkten Kontakt (direct touch) kann in einem solchen Setup zu einer anstrengenden und über einen längeren Zeitraum zu einer ungenauen Interaktion, auch unter dem Gorilla-Arm-Effekt bekannt, führen. Um diese Probleme zu überwinden und eine Interaktion zu erleichtern, stellen wir zwei Blick-basierte (gaze-based) Interaktionstechniken vor. Mit blickbasierten Interaktionstechniken wird die direkte Berührung nur für den horizontalen Bildschirm notwendig. Damit ein solches Setup richtig funktioniert, müssen sogenannte Switching Modes genutzt werden.

Die neu entwickelten blickbasierten Techniken heißen ITSS und ITOS. Indirekte Touch-Screen-Selektion (ITSS) ermöglicht es Benutzern, den Bildschirm mit welchem sie beabsichtigen zu interagieren, zu wählen, in dem sie diesen einfach ansehen. So wird eine absolute Umsetzung des auf der horizontalen Ebene durchgeführten Inputs auf der vertikale Ebene ermöglicht. Indirekte Kontakt-Objekt-Selektion (ITOS) markiert Objekte, die der Benutzer ansieht und nutzt das direkte Berührungs-Mapping. In drei Benutzerstudien untersuchen wir unsere beiden augenbasierten Interaktionstechniken und vergleichen Sie sie mit der traditionellen direkten Berührungsinteraktion.

Acknowledgements

First of all, I would like to thank Prof. Dr. Jan Borchers for giving me an opportunity to work on this great project at the chair, for introducing and bringing to me the fundamentals of HCI field and for giving an opportunity to publish the results of this thesis on the conference. Special and a lot of thanks to my supervisor Simon Voelker, who supported and helped me throughout the whole period of work on my master's thesis, who gave me needed and useful feedback and helped me to improve myself a lot. I also would like to thank my second supervisor Prof. Dr. Johannes Schoening for a useful feedback.

Besides, I would like to thank Chatchavan Wacharamanotham for additional feedback about the user study design and user study itself.

I am very thankful to everyone who supported me in any kind of assistance during this work: Rene Linden, Philipp Wacker, Rene Bohne, Jan Thar, Claude Bemtgen, Jan-Peter Kraemer, Thorsten Karrer, the folks from Media Computing Group, all participants from my very long and tiring user study and everyone I forgot.

Moreover, I would like to thank my family and friends who supported me during the whole time working on my thesis in different ways, without them I would not be at the point in my life I am now.

Conventions

Throughout this thesis we use the following conventions.

- The whole thesis is written in American English
- Independently of the real users' gender we will use "she" as a reference to a single person.
- In **Chapter 4 - "User Study"** we use the following abbreviations: p probability ($p = 0.05$ is called a significant level)

Chapter 1

Introduction

Desktop workspaces

Systems like Tognazzini's StarFire (Tognazzini [1994]) or the more recent BendDesk (Weiss et al. [2010]) and Curve (Wimmer et al. [2010]) prototypes combine a horizontal and a vertical surfaces into a single continuous desktop workspace. On such systems one can use direct touch interaction on both vertical and horizontal screens.

Gorilla Arm effect

Direct touch interaction, however, brings difficulties and inconvenience to the interaction with the systems which combine vertical and horizontal touch screens. The longer the user directly interacts on the vertical screen, the more exhausting becomes an interaction. As shown by Hammerton and Tickner [1966], the more joints of the human body are involved into the interaction process, the more complicated and tiring gets an interaction. An effect, known as *Gorilla Arm effect* , is a result of the tiring interaction, because the user has to hold her arm in the mid-air for a while.

Vertical screen suits better for an information overview

The development and changes in graphical interfaces moved computer input devices to the horizontal surface and left the function of showing an information to the vertical screen. It helps to decrease the neck muscle strain and pain, as addressed by Voelker et al. [2013]. This change in the graphical interfaces and, therefore, in the computer setup prevents users from bending over the horizontal surface in order to see an information. Users instead keep their neck straight by looking at the vertical screen, as in everyday human-to-human communication.

Fingers rest during an interaction on the horizontal surface

The standard computer setup gives users an opportunity to interact with a system, by resting their hands on the horizontal surface. Thus, the load on the human's hand is significantly decreased. It allows users to work in workspaces with vertical and horizontal input surfaces for a longer time, and with less fatigue.

Input devices

As far as users do not have to directly interact with a vertical screen in their computer setups, they need to find a way to give an input information to the system indirectly, e.g. to use input devices. Some of the commonly used input devices - mouse, trackPad, graphic tablet - have different mappings from the horizontal to the vertical screen. For instance, mouse is an example of the absolute mapping device, while trackpad and graphic tablet involve relative mapping. All of them have their advantages and disadvantages and can be used according to the tasks and users' needs.

Cumbersome interaction with a vertical screen on BendDesk and Curve

Input devices, however, are not the only way for information input. Users have an opportunity to use touch input, in order to interact in the interactive workspaces. For example, in the mentioned above multitouch desktop workspaces like BendDesk (Weiss et al. [2010]) or Curve (Wimmer et al. [2010]) users use a direct touch interaction. An additional bending surface and direct touch allow them to smoothly proceed from one screen to another. But an interaction on these multitouch systems has disadvantages and raises an interaction problem. The results of the studies, presented in the papers by Weiss et al. [2010] and Wimmer et al. [2010], showed that direct interaction with a vertical screen is cumbersome and gets less accurate after a lasting period of work.

Approach to ease an indirect interaction with a vertical screen

To overcome the problems of direct touch interaction, many researches have shown different methods to *indirectly* interact with a vertical screen or with screens which are out of reach (Schmidt et al. [2009], Turner et al. [2011], Turner [2013]). An interaction approach proposed by Schmidt et al. [2009] allows users indirectly interact with object on the vertical screen and comfortably manipulate objects with multi-touch input. Users no longer need to physically touch the vertical surface. Instead, all of their touches from the horizontal surface are mapped to the vertical.

Switching mode problem

An indirect touch approach works great until the user wants to use the horizontal screen also as an output. It leads to the *the switching mode problem* and rises the question: should that touch go directly to the horizontal display, or indirectly to the vertical screen?

Ideas behind proposed indirect touch interaction techniques

In order to clarify the issue of choosing an interaction surface, and give user a clear understanding where her touch is mapped to, overcome the fatigue problem, and increase the comfort in interaction, we present two interaction techniques that use gaze to choose where each touch should be directed. The first technique is Indirect Touch Surface Selection (ITSS). User's gaze determines the target surface and her touch is directed there, using an absolute mapping from the touch surface to the output screen. The second technique is called Indirect Touch Object Selection (ITOS). Here, user does not use the gaze to just determine the right display, instead she selects an object on the screen just by looking at it. Thus, all the touch events are directly mapped onto this object. If the touch is mapped to one object, it stays there until the user releases her finger from the touch screen. ITOS technique follows the principle "gaze suggests, touch confirms" as in work by Pfeuffer et al. [2014]. User selects an object of interest and confirms that selection with a touch. To sum up, the key idea of our approach is to use eye-gaze as a base to switch between the screens in order to ease an interaction.

Study results

In the user study, conducted on the system shown on Figure 1.1, we investigate and compare two gaze-based interaction touch input techniques. They combine direct and indirect touch, with a baseline direct touch (DT) condition, as in an office workspace environments like BendDesk (Weiss et al. [2010]) and Curve (Wimmer et al. [2010]). Our results show that indirect object selection technique surpasses the other two techniques in tapping speed on the vertical and horizontal surfaces, and dragging only on the vertical surface. Considering the tasks which involve cross dragging, our results show more complex dependencies and correla-

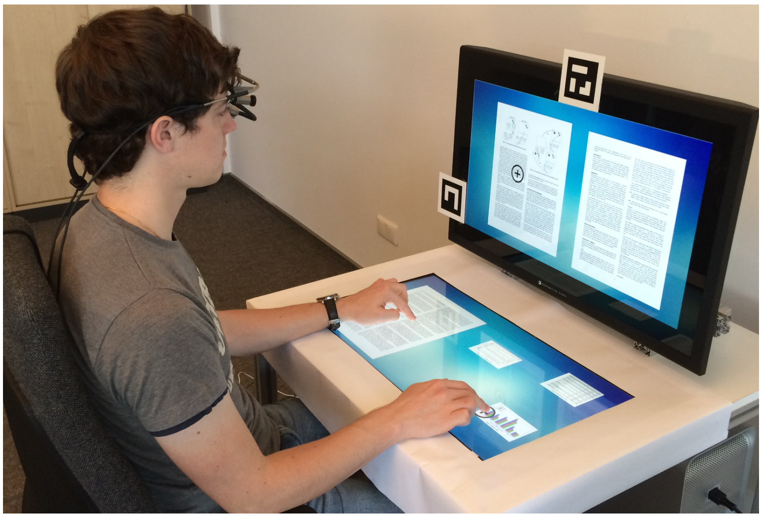

Figure 1.1: One can interact on the horizontal surface using direct touch and on the vertical surface using indirect touch with a help of user's gaze.

tions. We describe in details the new gaze-based interaction techniques, present the statistical evaluation of user's performance with these techniques, give an application scenario of where they can be used, and give an overview of how such a system can be extended to the multiple touch screen interaction environment.

Thesis contribution

This thesis has three main contributions:

1. We present two gaze-based interaction techniques with direct touch as a baseline to allow users to switch between the direct and indirect touch interaction on the horizontal and vertical screens.

2. We provide an empirical evaluation of the proposed techniques in comparison to the traditional direct touch interaction.

3. We present a real life scenario application to show how our developed techniques can be used.

Structure of the Thesis

In the following, we outline the structure of this thesis and give a short overview of each chapter:

Chapter 2 - "Related Work" In this chapter, we present the related work overview of the work done in the areas: "Touch Interaction", "Interaction on Horizontal and Vertical Screens", "Systems with Multiple Surfaces", "Gaze-based Interaction".

Chapter 3 - "Interaction Techniques" In this chapter, we describe two- and three-state interaction models and new proposed gaze-based interaction techniques.

Chapter 4 - "User Study" In this chapter, we outline the pilot and main user studies with 3 conducted tasks. In the subsections we describe the task users had to perform, the results and discussions.

Chapter 5 - "Application Scenario" In this chapter, we present an application suggestion of how our techniques can be used, describe the general concepts and the implementation details for the developed application.

Chapter 6 - "Summary and Future Work" In the last chapter, we summarize the content of the work presented in the thesis, give an outlook, where and how ITSS and ITOS can be used in the future, and discuss the possibilities of further improvements.

Chapter 2

Related work

In the following chapter we outline the related work done in gaze and touch research areas. The chapter is split into 4 subareas: ”Touch Interaction”, ”Interaction on Horizontal and Vertical Screens”, ”Systems with Multiple Surfaces” and ”Gaze-based Interaction”.

2.1 Touch Interaction

Fatigue and exhaustion during direct physical interaction with vertical screen

It was shown by Weiss et al. [2010] that interacting on the horizontal surface can be done faster and with less fatigue than on the vertical surface using direct touch interaction. Moreover, exhaustion consideration by Weiss et al. [2010] and Wimmer et al. [2010] for BendDesk and Curve (Figure 2.1) report the arm fatigue after a long-term interaction on the vertical screen and suggest resting an elbow on the desk to reduce it. In his design considerations Wimmer suggested tilting a vertical screen for $15°$ as one more way to reduce the fatigue and physical load, so that arm can lean on the vertical screen.

Approaches to reduce users’ neck pain and strain

Schmidt et al. [2009] compared input on a tabletop display with an indirect condition, where the horizontal surface was used as an input. He underlined that the arrangement of the input surface strongly influences the comfort and flu-

Figure 2.1: Curve is a digital desk concept that blends a horizontal and a vertical interactive surface (Wimmer et al. [2010])

idity of the interaction, by evaluating different positions of the input surfaces. His intends were directed towards the reduce of users' neck pain and strain, in order to avoid the bending over the horizontal surface.

The problem of input indication

As far as indirect input was suggested as an alternative approach to the direct touch, the problem of input indication has arisen. Therefore, Schmidt et al. [2009] mapped a shadow from the user's arm, which appears on the horizontal surface, to the vertical. However, according to his evaluation results, this indirect touch approach did not solve the problem of fatigue, because users had to hover their hands over the horizontal surface for a long time.

Combining touch screen and mouse interaction

Other researchers were trying to combine already existing input devices with touch input. For example, Bi et al. [2011] combined mouse and touchscreen, in order to extend the interaction surface, and compared different zones of interaction. He concluded that tasks were performed faster in the areas closer to the keyboard and mouse, because of the short travel distance.

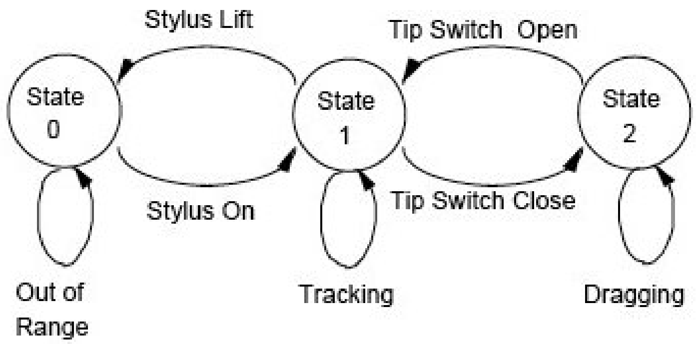

Figure 2.2: Three-state model of graphical input (Buxton [1990])

Evolving of *tracking state* for indirect touch interaction

Indirect interaction on the vertical display involves the usage of the *tracking state*. It allows users to track their current position on the indirect surface. They need to know where their touch is currently mapped to and what actions they have to perform, in order to reach the target or plan the dragging movement. This problem of the *tracking state* was picked up by Voelker et al. [2012] and Buxton [1990] (Figure 2.2). This state allows users to touch the surface without provoking an unintentional interaction with the displayed objects and, therefore, exclude the situations of unwilling interaction. *Tracking state* provides users with a clear indication of where their touches are, and gives an awareness of the current interaction.

Therefore, in our gaze-based techniques for indirect interaction (introduced and described in details Chapter 3 - "Interaction Techniques") we decided to integrate switching modes between the vertical and horizontal screens, and include the *tracking state*. With these additions we try to reduce the hand movement in the mid-air. As far as our switching modes use the eye-gaze, in the section 2.4 of the related work we outline the problems and aspects of eye-gaze based interaction in the combination with other input modalities.

2.2 Interaction on Horizontal and Vertical Screens

Vertical and horizontal surfaces have different purposes and, therefore, can be used differently in the setups, which combine both of them. Recently many researchers were trying to study certain characteristics of horizontal and vertical surfaces, describing and analyzing their benefits and drawbacks.

Interaction on the horizontal surface is inconvenient, but suitable for annotation

As shown by Morris et al. [2007], horizontal setup is suitable for an annotation tasks more than vertical. In the same study he also analyzed the features of the vertical and horizontal surfaces for reading and annotation tasks. Vertical screen was used for the presentation of the output information, with mouse and keyboard as input devices. Horizontal setup consisted of two pen-enabled displays positioned horizontally on the table. Interaction on the setups that contain only a horizontal surface is rather inconvenient, therefore, it was proposed to use the system which combines both vertical and horizontal surfaces.

Tilting the vertical surface helps to increase a comfortability of usage

In the later study by Morris et al. [2008] it was shown that attempts to position the horizontal display, mouse and keyboard in a way possible for comfortable interaction still cause problems and inconveniences for users. Users had problems observing both horizontal and vertical surfaces at the same time, by perceiving the surfaces as they were not connected. Supporting the idea of Wimmer et al. [2010], they found out that vertical screen should support tilting, in order to increase a comfortability of usage.

Vertical display - showing an information, horizontal - "private" area

Another mock-up study by Müller-Tomfelde et al. [2008] showed that tilted displays were rather preferred by users. Except that, they showed that vertical display is suitable for displaying an information, which also can be seen from the larger distances. The horizontal surface was used as a "private" area, because objects on the horizontal screen can only be seen by the people who are close to the surface.

Perception of the objects is different on different displays

Wigdor et al. [2007] studied the perception of the graphic elements separately on the horizontal and vertical surfaces.

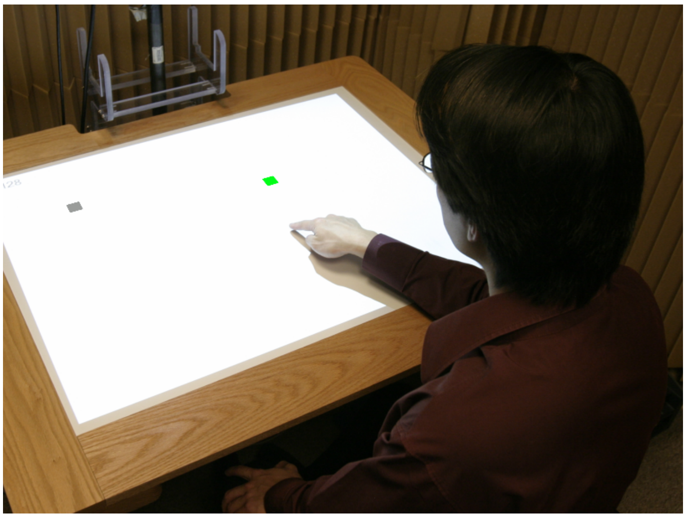

Figure 2.3: In Expt. 1, participants used a finger or a mouse to select and drag targets. In Expt. 2, they used either two mice or two fingers to perform a symmetric bimanual docking task (Forlines et al. [2007]).

The outcome showed that objects are perceived differently for different position of the surfaces. The perception of the object on the vertical display does not change from the different angles, in comparison to the horizontal surface, where the perception of the object depends on its position on the surface.

Hand pose changes during the interaction with different displays

Forlines et al. [2007] (Figure 2.3) discovered that the position of the users' hands change with the change of the interaction point on the combined display system. Interaction with a vertical screen does not change the pose of the hand during the interaction. However, the hand pose is different for objects more distant from the user, in comparison to the closer ones.

2.3 Systems with Multiple Surfaces

As far as we are not restricted only to two screens or surfaces in interactive workspaces, we would like to underline the extensions and aspects of the multiple screen interactions.

Vertical screen - information overview, horizontal - input

Systems which combine multiple interaction surfaces are widely used for collaborative and remote interaction workspaces. And as was mentioned before, vertical screen is used for an overview of the shared information and horizontal surface is used for information input. Moreover, horizontal surfaces support the side-by-side and face-to-face collaboration, where users can sit physically close to each other and interact with a system.

Possible ways of transferring the content between the screens

In the systems with multiple horizontal and vertical displays it is not only important to achieve a suitable and comfortable interaction and information overview, but also to afford a transfer of the data between the screens. Rekimoto and Saitoh [1999] were exploring the possible ways of smooth interchange of the digital information between the portable computers, tables, wall displays, and physical objects. By projecting the displays on the walls and tables, users could have a spatial extension of their portable computers. Knowing the spatial relationships between the computers situated in the room, users could drag the objects between them, using a technique called *hyperdragging* . The idea behind this approach was to use a camera-based object recognition system, which allowed users to easily integrate their portable computers with a pre-installed environment.

Multi-modal collaboration system

A platform DIGITABLE presented by Coldefy and dit Picard [2007] allows to fill the gap between a co-presenter and a distant interaction. In the environments with multiple users the system provides a smooth transition and visualization of the participants' gestures, by showing whether she is moving an object or has intends to do so. Moreover, DIGITABLE combines multiuser tactile feedback, video-communication with eye-contact, a specialized sound and voice transmission. This system contributes to the efficiency of collaboration, by allowing users to talk and coordinate during an interaction.

Extension of working environment with walls and tables

Everitt et al. [2006] presented another multi-user interaction system, which allowed people to access and share the data. This system supports mobility and micro-mobility of electronic content between tables and other devices. They

Figure 2.4: Different setups of FLUX , as a drafting table (left), as a discussion table (middle) and as whiteboard (right) (Leitner et al. [2009])

found out that an augmented space would ideally include both tables and walls, because people prefer to use environments, to which they are already used to. The functionality of the possible devices was split, and the results showed that personal devices suit best for content creation: a table - for content organization, providing a shared space for users' collaboration, and wall - for content representation. Parallelism is a needed part of interaction process, because sometimes users want to work independently and in parallel, and sometimes - collaboratively.

Vertical screen should be placed in front of the horizontal

Wigdor et al. [2006] in his study explored the suitable position of the horizontal and vertical screens in the setup with two screens. His findings showed that vertical screen should be placed right behind the horizontal, in order to give users an opportunity to observe both screens at the same time. Users preferred to orient their control space - an area used by a user to provide input to the system - between 0 and 90 degrees, or generally, on the east.

System with one screen which could be used as vertical and horizontal

In the work by Leitner et al. [2009] it was proposed a completely new approach of combining horizontal and vertical screens - adjust the system according to the task. So, in the FLUX system, which consists of only one multi-touch surface, users could tilt the screen to the comfortable position and change the horizontal screen to the vertical and vise versa. The idea was to create a flexible system that could be used for different tasks and, therefore, users could benefit from the advantages of horizontal and vertical screens (Figure 2.4). FLUX could be used either as a sketching board, or as an interactive discussion table, or as a digital presentation whiteboard.

2.4 Gaze-based Interaction

Eye-gaze support is the crucial part of the proposed eye-gaze based techniques described in this master's thesis, therefore, we outline the possibilities and extensions of the current eye-gaze usage.

Nature and physical properties of human eyes in target acquisition tasks

Eye focus selection is fast, natural, requires little effort and provides a context for other forms of interaction (Smith and Graham [2006], Stellmach and Dachselt [2012]). The eyes usually acquire the target of interest before the arm's muscles are involved into the interaction process (Zhai et al. [1999]). Therefore, Jacob [1990] suggested the principle "what you look at is what you get". Human have an opportunity to experience this principle everyday, receiving a major portion of information via the visual channels. Before executing any command users tend to look at a target (Shell et al. [2004]), what can be used as a good focusing approach for windows or screen targeting. Users, however, can interact with objects without looking at them, using a tactile feedback. But in our system we do not use any tangibles or other objects which help to estimate "where we are".

Combination of eye gaze and mobile devices

Turner [2013] combined an eye gaze based interaction with mobile devices and evaluated the techniques which allow users to transfer the data from a hand-held device to the distance display and vise versa (Figure 2.5). The techniques for interaction in such conditions are already outlined by Turner et al. [2014] and Turner et al. [2011]. His study showed that manual interaction outperformed the gaze-based positioning. This finding leaves the gaze input interaction a supporting place, e.g. switching the mode or outlining the focus of the interest in the manual interaction.

Dwell time approach

Turner et al. [2011] showed that the techniques with dwell time action confirmation appeared to be slower in comparison to the touch-based confirmation. This was also shown not only for workspaces, but also for hand-controlled device interaction. As far as dwell-time interaction involve concentration on a particular point on the screen for a long time, it causes the fatigue and leads to inability for a long

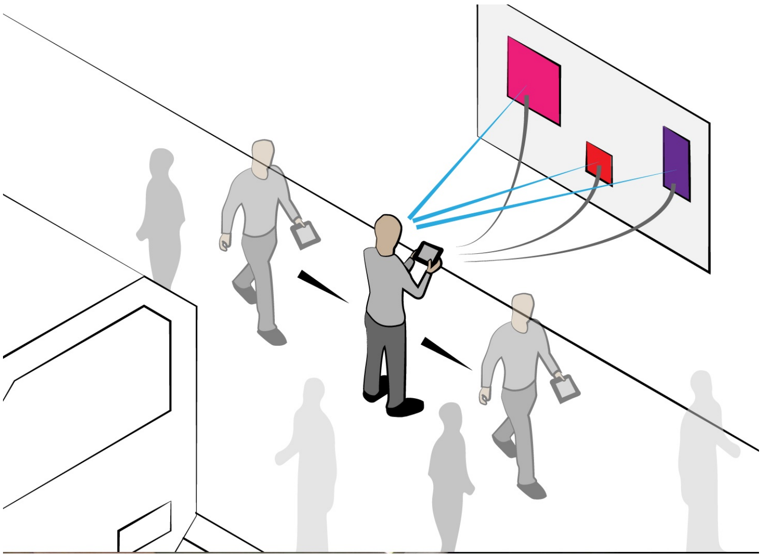

Figure 2.5: Users acquire content from the distant displays (Turner [2013])

time usage.

Switching modes in the gaze-based interaction

Pfeuffer et al. [2014] showed that switching modes can be used in the combination with gaze-based input. They combined eye-gaze tracking with a single tabletop and placed users' touch points at the point of the user's gaze focus, using a principle "gaze suggests & touch confirms". Gaze-touch was designed to complement traditional direct touch on multi-touch surfaces.

Separation of hand and target

Combination of touch and gaze indirect interactions spatially separates the hand from the target. This separation can be considered from two viewpoints, as shown by Pfeuffer et al. [2014]. First, users can reach and select any object on the surface without physically moving the hand. Second, the same target can be manipulated from different positions on the surface.

Multiple object selection

Their techniques allowed multiple object selection, by supporting simultaneous input from multiple points. Users had to look at each target and perform a touch down. This approach allows to map multiple objects to each finger and perform a simultaneous manipulation. It also allows users to overcome the reachability problems and select objects,

Figure 2.6: Users select by gaze, and manipulate with multi-touch from anywhere (Pfeuffer et al. [2014])

which are out of the physical hand reach, as well as select objects which are distant from each other.

Gaze-touch results

The results of their studies prove the advantages of gaze-based input such as speed, reachability of the distant display, decreased fatigue and physical mid-air hand movement. They also outlined a space design analysis of combined touch with gaze input, and several applications, which explore how these techniques can be used along-side (Figure 2.6).

Double role of the gaze

Problems such as *the double role of gaze*, also known as the Midas Touch problem , inaccuracy and ineffectiveness, using the gaze for direct manipulation of the objects were outlined by Stellmach and Dachselt [2013]. Stellmach et al. [2011] addressed the existing problems which one can face while using the gaze input method. However, it was indicated that gaze usage is possible, as far as certain design considerations are taken into account.

Head-tracking and mouse combined interaction

Head-tracking in the combination with mouse was studied by [Ashdown et al., 2005]. Even though the interaction time for such an interaction increases, it was preferred by most of the users in his study, due to the less physical distance needed to travel. Head motion is rather stable in compari-

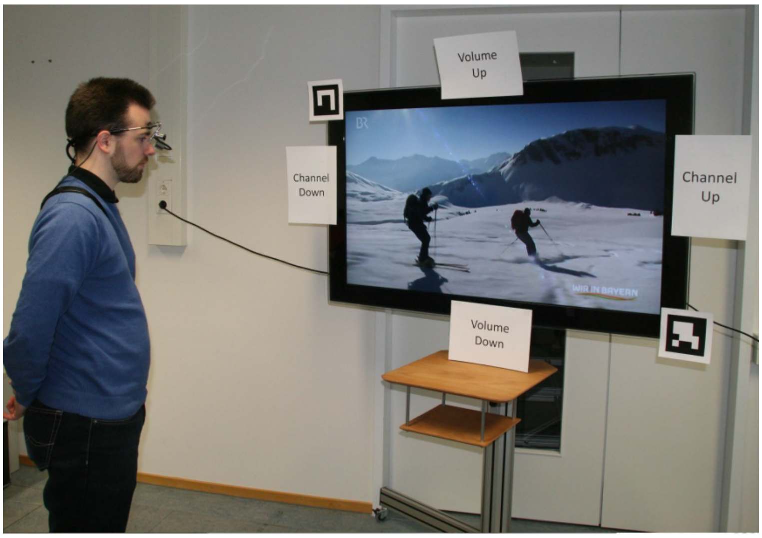

Figure 2.7: Experimental setup for video controlling (Breuninger et al. [2011])

son to the eye movement, and can be used for indication of the user's attention focus.

Remote targets acquisition

Nancel et al. [2013] investigated high precision pointing techniques for remote acquiring of the targets, and concluded that head orientation for coarse control for touch and cursor for precise selection was the most favorable and successful technique.

Gaze TV control

Mardanbegi and Hansen [2011] presented a scenario where they could interact with a TV and a computer screen, located in different places of the home environment, using a mobile phone and eye tracker. Breuninger et al. [2011] also used a gaze-based interaction as a supporting tool for TV-set and music player control (Figure 2.7).

Combining eye gaze and mouse

Zhai et al. [1999] presented techniques that combined mouse an gaze input called MAGIC (Manual and Gaze Input Cascades), in order to increase the speed of target selection task. Fono and Vertegaal [2005] combined gaze input with hot keys and mouse activation, and presented 4 different attentive windowing techniques for selection. They concluded that interaction with key activation was efficient and effective and, on average, as twice as fast as mouse or hot keys. The technique with key activation was the most preferred among the participants and was the fastest among the other techniques for focus window selection task (about 72% faster than manual conditions).

We extend an existing interaction model and describe interaction techniques which allow screen and object selection.

Chapter 3

Interaction Techniques

In this chapter, we state problem of switching modalities, describe new proposed gaze-based interaction techniques and interaction models.

3.1 Problem of Switching Modalities

Switching mode problem

Traditional approach to work with systems, which combine horizontal and vertical touch screens, is direct touch on both surfaces. We, however, want to use the indirect touch instead, because direct interaction with the vertical surface is cumbersome and leads to fatigue. User creates touch on the horizontal surface in front of her, and the output is displayed on the vertical screen. This approach works great until the horizontal surface is used only as an input. As soon as user wants to use the horizontal surface as an output, we got the switching mode problem. Thus, we need to know, where and how our touch is mapped.

Switching between modalities

As shown in the previous chapter "Related Work", eye-gaze has a potential to complement touch interaction, and in particular on the systems with vertical and horizontal touch screens. Adding gaze to the direct touch interaction helps solving the switching problem. Eye-gaze can be used to choose the interaction modality, and allow users to use both direct and indirect touch. If the user looks at the horizontal screen, she uses the direct touch on it. And if she looks at the vertical display - indirect touch. Therefore, we need techniques which allow switching between these two modalities.

Tracking state

Using the indirect touch on the vertical display, however, requires an additional *tracking state* in the touch model. User has to know the position of her touch on the vertical screen, in order to perform a successful manipulation and interaction. Unawareness of the mapping might cause an unwilling interaction.

Absolute mapping

As far as absolute mapping of touch from the horizontal surface to the vertical is simple and easy to understand, especially for multi-touch interaction, we decided to use this mapping in our Indirect Touch Screen Selection (ITSS) technique, which we describe in the next section. Other mappings, in our opinion, would lead to confusions.

3.2 Indirect Touch Screen Selection Interaction Technique

Touch is remapped to the vertical screen if user looks at it

Indirect Touch Screen Selection (ITSS) technique combines both direct and indirect touch approaches. If user's gaze is directed towards the vertical screen, her touch is mapped to the vertical screen, if elsewhere - to the horizontal. For this technique the default area of interaction is the horizontal surface, e.g. if user's gaze is not directed towards the vertical screen, then it's mapped to the horizontal.

Initial point of gaze focus

We only use the information where the user is looking at the very moment she creates new touch. As soon as that touch is mapped to one surface, it stays there until the user ends up that touch interaction by lifting her finger again. With every new touch input to the horizontal surface and eye-gaze direction user selects the screen from the start. This approach solves the switching mode problem and gives user an awareness of where her touch is mapped to.

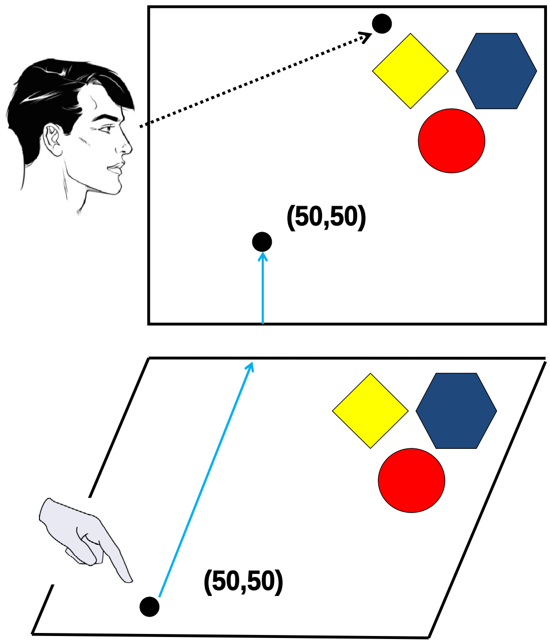

Figure 3.1: ITSS mapping example. Blue line indicates the transfer of the touch. User looks somewhere on the vertical screen and touches the point (50,50) on the horizontal screen, which is mapped to the same point on the vertical.

Example of touch remapping using ITSS

As far as the initial point is already mapped to the vertical screen, the user is willing to continue interaction there. Thus, the next issue, which comes up, is how to map the touch from the horizontal surface to the vertical. For this purpose we used absolute touch mapping. This can be easily done, because the size ratio of the screens in our setup is 1:1. So, the user can be aware of where her touches are, what is especially important in the multi-touch interaction. If, for instance, the touch on the horizontal surface appeared at the point with screen coordinates (50, 50) and focus was on the vertical screen, then the touch will be mapped to the point with coordinates (50,50) of the vertical screen (Figure 3.1).

Extreme interaction situation

Moving head aside from the screen, quick switching be-

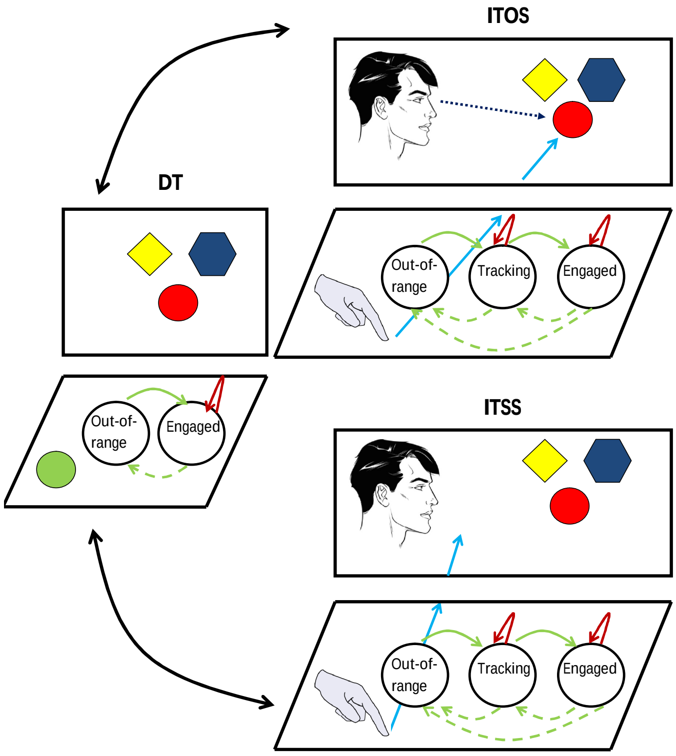

Figure 3.2: DT - direct touch. ITSS - indirect touch with an absolute mapping of the input touch. ITOS - indirect touch with a relative mapping of the input touch. Blue line indicates the transfer of the touch.

tween the screens by moving the eyes, or any other kinds of distractions do not cause any problems while using ITSS. As far as touch input stays still on the horizontal screen, the mapping stays still too. If, for example, the current touch is on the vertical screen and user does not release her finger from the horizontal screen, then touch will stay on the vertical screen. Even though the user is distracted or moves her eyes very quickly between the screens, interaction still stays there. But if user releases the finger and looks aside, then touch mapping goes to its default horizontal screen, and only looking at the vertical screen allows her to jump back.

Each touch point which is mapped to the vertical screen is in the *tracking state*. Interaction in this state allows user to move the cursor above the objects on the screen without moving them. The cursor represents a touch point and stays on the surface until the finger is released from the input surface. This allows users to manipulate the objects bimanually and with multiple fingers. In order to switch to the *engaged state* user has to execute *lift and tap* operation as suggested by Voelker et al. [2013]. This interaction process is presented in Figure 3.2.

The benefit of *tracking state* in ITSS

3.3 Two- and Three-state Interaction Models

At the moment, the default way to interact with interactive desktops is using a traditional direct touch approach, where users can physically move their hands and acquire the objects of interest. While using direct touch in the setups with multiple horizontal and vertical screens users do not need a special switching modes to select one screen or another. Their "switching mechanism" is a physical contact with a screen of interest, so they know, when they touch a particular screen and where the touch is mapped to. Direct touch interaction allows users to simultaneously and bimanually interact on both screens. They are aware of where their touch is mapped to, and what objects they are currently interacting with, because interaction focus can be only present on one screen at the time.

Current methods of interaction with interactive desktops

Direct touch interaction follows the two-state touch model (Figure 3.3). It has two states: *out-of-range* and *engaged*. If the touch input is present on the surface, then current system's state is engaged. If the touch is not present, then the system is in the out-of-range state.

Two-state model in direct touch interaction

However, as mentioned in the previous sections, interaction with horizontal and vertical touch screen using direct touch is tiring and uncomfortable. Therefore, turning workspaces with multiple screens and direct touch interaction technique into the long-term usage environment is

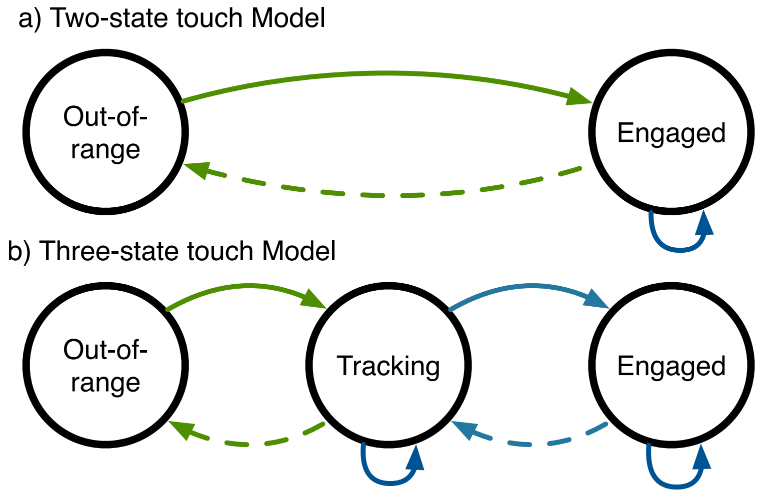

Figure 3.3: Two- and three-state model diagrams (Voelker et al. [2013])

rather complicated. One of the possible ways to solve these complications is extending the touch interaction model to three-state, which means turning the system into indirect touch system (Figure 3.3). The interaction model behind this approach has three states: *out of range*, *tracking*, and *engaged* [Buxton, 1990].

Idea of remapping of touches from the horizontal to the vertical screen

To overcome the mentioned above problems of fatigue and distant screen reachability while using the direct touch, we developed two indirect touch gaze-based interaction techniques and integrated the three-state interaction model. While using those indirect touch techniques users do not have to physically touch the vertical screen anymore. All of the touch input they perform on the horizontal screen in some cases stay on the horizontal screen, and in some, it is mapped to the vertical screen. Those cases are determined by the users' gaze. We decided to use eyes for between the screens switching, because of its speed and naturalness in everyday human-to-human interaction.

Gaze-based switching modes help users to switch between the interaction surfaces

In order to give users an awareness of where their touch is and what objects are currently under their control, we present gaze-based switching modes. The idea of using eyes as focus of attention is not new and was mentioned before in the works of Zhai et al. [1999] and Jacob [1990].

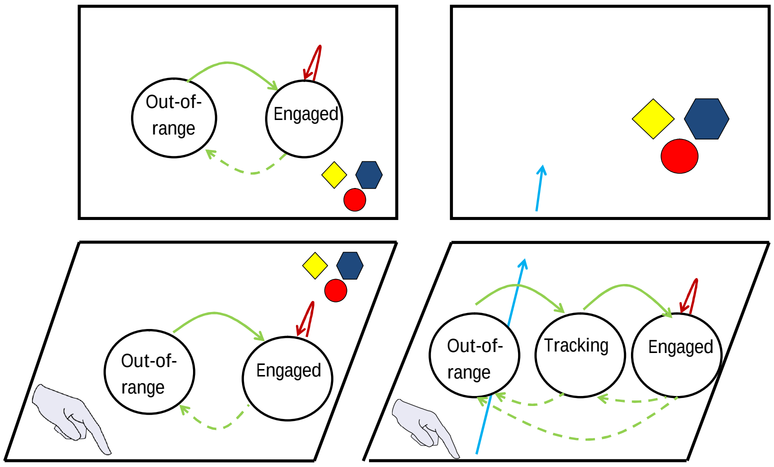

Figure 3.4: Schematic visualization of direct and indirect touch models. Blue line indicates the transfer of the touch.

Users usually look at the objects before starting an interaction with them. After starting an interaction the user's focus might change, but they still can continue an interaction, because users are aware of where their touch is. We followed this principle and implemented two techniques which allow selecting screens and objects of interest. According to the three-state model, users still can use multiple fingers and two hand interaction on both screens, even though their physical touch is present only on the horizontal surface (Figure 3.4).

Tracking state gives a permanent feedback about the position of the touch

While using an indirect touch technique user has to estimate the area on the horizontal screen, which is mapped to the vertical. Moreover, within this interaction model there might be situations, which lead to the unintentional manipulation of the object. With direct touch, users hit the objects directly, but indirect touch requires a *tracking state* to give a constant feedback of where the touch is, and to avoid unwilling manipulation of other objects. Using such a "cursor" from the tracking state as a representation of the user's touch, the user can move to the position on the surface she wants. When the cursor reaches the target she can execute a tap action or decline it. Thus, the horizontal screen turns into a single input surface, while vertical stays as an output only.

Avoidance of the unintentional interaction while using ITSS and ITOS

Our interaction techniques are used only for non-critical tasks, such as surface or object selection. If during focusing on the surface or object the touch is not present, nothing happens and unintentional manipulation is avoided. Moreover, as far as touch stays on the horizontal surface after selecting a screen or an object, the direction of users' gaze can be changed, because eye-gaze is used only for the initial point mapping. The situations with ringing phone, looking aside or some other distractions do not ruin users' interaction process and she can get back to the interaction when the distraction is gone, or even when it is still present.

In the next section we present and describe in details our next gaze-base indirect touch technique ITOS.

3.4 Indirect Touch Object Selection Interaction Technique

Touch is remapped to the vertical screen if user looks at it

Indirect Touch Object Selection (ITOS) technique allows users to select objects of interest using their eye gaze. Similar to the described before ITSStechnique, when the user's eye gaze is directed towards the vertical screen, all touch is mapped to that screen, if not - touch stays on the horizontal surface. Again the default screen is horizontal, e.g. if user is looking somewhere else and touch is performed, except for vertical screen, her touch is mapped to the horizontal.

Selection of the objects in the gaze area

With this technique user can not only select the screen she is currently focusing on, but also an object close to the focus. The initial touch is transferred to the object which is in the user's focus. As far as human eyes are involved in the constant movement, it is complicated to achieve a permanent fixation on the single point of interest, what is also known as *jittering*. Other factors which might come into play are tracking inaccuracies, e.g. calibration software and hard-

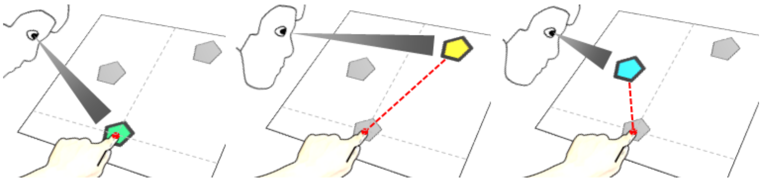

Figure 3.5: More expressive input from the same touch position: three examples of users touch on the same touch position, but each time manipulate a different target (Pfeuffer et al. [2014])

ware itself are not perfect, and bring inconsistency into the system. Influences of changes in pupil dilation or the role of the dominant and non-dominant eye might be another factors. In order to overcome eye jittering problem, we used a snapping approach, similar to the one proposed by Pfeuffer et al. [2014] (Figure 3.5). Although the *jittering* is present, we can estimate an area where users' gaze stays during the fixation on one point. Let's say we are playing darts and target board on the wall is our target, but our arrows do not always hit the target and might hit some parts of the wall close to it. According to this imprecision in targeting we extend the estimation area, including to the target area the area around it. The same principle we use in our approach to overcome *jittering*, by increasing the radius of object "selector".

The concept of ITOS interaction technique

The user's touch is translated to the object where she is looking at. Before the touch is performed, an object gets highlighted, indicating the focus point of the user. If the object is highlighted and touch is performed anywhere on the horizontal surface, the cursor appears in the center of the highlighted object and interaction switches to the *engaged state*. After this operation a user can look outside the vertical screen, but her touch will stay on that object till she releases the finger from the input surface. If the user touches the input surface and looks at the object, the system estimates an area, where user's focus was concentrated for the last 50 ms of interaction. If inside this estimated area only one object was present, then it gets highlighted and becomes ready for selection. In cases of multiple objects presented in one area close to each other, the system estimates not only the focus area, but also the center of it.

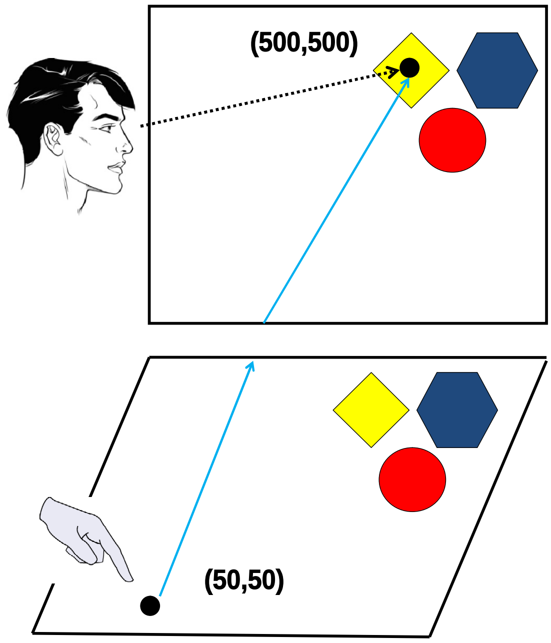

Figure 3.6: ITOS mapping example. Blue line indicates the transfer of the touch. The user looks at the object on the vertical screen, touches anywhere on the horizontal screen and her touch is transferred to the object on the vertical screen.

The closest object to the center of that area gets highlighted. By using this approach of estimating the area of concentration, we overcome the problem of eyes jittering, neglect the blinking of the object caused by eye jittering and achieve a stable interaction.

Example of touch remapping using ITOS

For example, when an object at the position (500,500) in the vertical screen coordinate system is highlighted, e.g. it is in the user's focus, and user touches the horizontal surface at the point (50,50) in the horizontal screen coordinate system, then cursor is going to appear at the point (500,500) of the vertical surface - position of the highlighted object (Figure 3.6). So, the mapping of touch from the horizontal to the vertical screen is relative.

The main difference between ITSS and ITOS techniques is the absence of the *tracking state* in ITOS. Before a user looks at the object the current state is *out of range*. As far as his gaze is concentrated on the object of interest the object gets highlighted, but the current state does not switch to the *tracking state*, as far as no touch on the horizontal surface was performed. If object stays in the user's focus and the touch is performed, then state is changed to *engaged*. This interaction model prevents user from unintentional interaction. If the user's gaze is not directed towards the vertical screen, then interaction is done by traditional direct touch technique.

The difference between ITSS and ITOS techniques is the absence of the *tracking state* in ITOS

Chapter 4

User Study

We conducted three experiments in order to evaluate our proposed gaze-based indirect touch techniques, and compared them to the traditional direct touch. We designed three tasks: *tapping*, *dragging* (dragging an object across the horizontal or vertical surfaces) and *cross dragging* (dragging an object from vertical screen to the horizontal and vise versa). In our user study we focused only on the single finger interaction with the user's dominant hand. In the future this can be extended to the multiple finger and bi-manual interaction, what will be addressed in detail in the section "Future Work".

The purpose of the user study is to evaluate the proposed techniques

We conducted 3 experiments in order to answer the following questions:

1. Which technique was preferred for the indirect touch approach?
2. Which technique performed better in the factor of *speed* and *accuracy*?
3. Which technique is the best for a distant screen interaction?

All three experiments were conducted using the same set of participants, same setup and overall procedure.

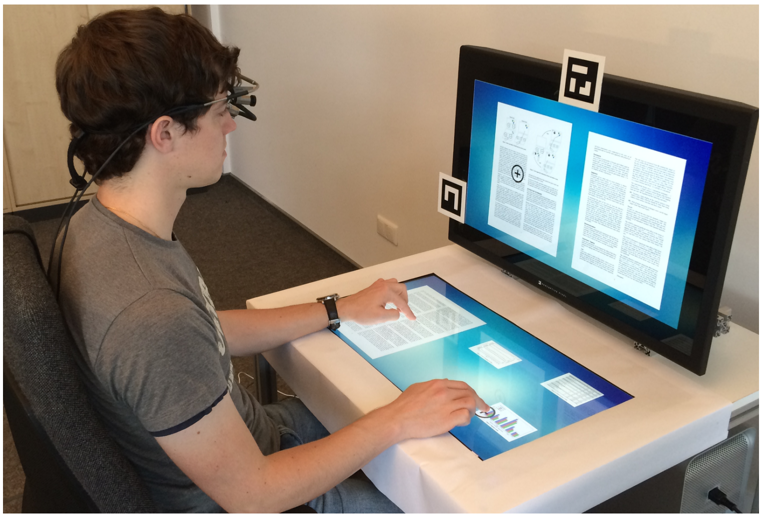

Figure 4.1: Pilot study setup

4.1 Pilot Study

The purpose of pilot study is to understand how users focus

Before conducting the main user study to evaluate two proposed gaze-based interaction techniques, we conducted a pilot user study to observe how users *focus* while they interact with different input devices. The purpose of this study was to determine whether our idea of creating indirect touch technique by placing a cursor directly on the object or screen of interest would take place. Thus, if users are looking at the object which they are willing to approach, then it makes sense to place a cursor on that object. We also wanted to see whether users look at the input devices during an interaction.

Pilot study description

We run a study on the system which combined one vertical and one horizontal display (the same setup as in Apparatus section below and shown on the Figure 4.1) We tested four different input devices: direct touch, mouse, trackPad, and touch system with absolute mapping (all touches from the horizontal screen were absolutely mapped to the vertical screen - similarly to ITSS technique). We recruited 5 users (2 female and 3 male). All of them had an experience using mouse and four of them with using a TrackPad.

Task description

Users were instructed to perform a tapping task. They had to tap on the objects which appeared only on the vertical screen. During the study we were recording the videos of users interacting with the system. Afterwards we analyzed the videos and estimated how much time users spent looking at their input devices.

Pilot study results

The results showed that the focus of 2 participants was always on the vertical screen and for other 3 participants the focus was about 99% on the vertical screen. These outcomes allowed us to continue with developing ITSS and ITOS interaction techniques.

4.2 Participants

As far as our pilot study showed a promising results, we decided to conduct the main user study and recruit new participants. A total of 14 participants (9 male and 5 female) aged between 23 and 36 (mean age 27) took part in our experiment. 12 of the users were right handed and 2 were left handed. As mentioned above, during the experiments users were allowed to use only their dominant hand. An average duration of all three experiments was about 1.2h.

After each of the user study task the participants were asked to answer the questions from the user study questionnaire (appendix A - "User Study Questionnaire") concerning their fatigue experienced during the task, preferences among the techniques for different tasks, and their own opinion and feelings about the speed of the techniques. The tiring effect was ranged from totally agree and totally disagree. The answer sheet had 5 cells for giving an answer. For the technique preferences users could directly select the name of the technique. All of the users were familiar with direct touch interaction. None of them had an experience using eye-gaze tracker.

Figure 4.2: Tabletop of the system. Horizontal surface.

4.3 Apparatus

The setup consists of two touch screen displays

The setup presented for the user study was the same as shown on Figure 4.1. As a horizontal screen we used a capacitive touchsensing 27″ Acer Touch display embedded in a custom made table at a height of 72 cm following ISO9241-5 (Figure 4.2). For the vertical screen we used a 27″ Perceptive Pixel display, which was placed 55 cm from the edge of the table. Both displays had the same resolution of 2560 x 1440 pixels and size of 597 x 336 mm. Both displays were connected to a Mac Pro running the software for the experiments. The effective touch frame rate for both displays was set to 60 Hz.

We did not test the setups with different angles for the vertical screens, because we were interested in the standard computer-based setup.

Users' gaze was determined by the Ergoneers *Dikablis Glasses*

Users' gaze was determined by the Ergoneers *Dikablis Glasses* (Figure 4.3). The *Dikablis Glasses* is a head-mounted eye tracking system and allows, in particular, to detect the position of the user's gaze in a visual marker coordinate system. Two markers were placed around the vertical dis-

Figure 4.3: Dikablis Glasses by Ergoneers.

play, as shown on Figure 4.1. The coordinates of the user's gaze in the *markers'* coordinate system are used to calculate and estimate the user's focus point. The conversion accuracy of the gaze coordinates into the pixel coordinate system for vertical screen is about 1.5 cm (63 px). The effective frame rate of the eye tracker was also set to 60 Hz.

Two-staged calibration process

The eye tracker calibration process had two stages. The first one included standard general calibration procedure provided by Dikablis software, which takes about 30 seconds. The second one was an additional self-written inner software calibration, which we were running before each gaze-based experiment. The second stage of the calibration allows to achieve much more stable gaze to pixel conversion for a longer usage period.

4.4 General procedure

Each of the experiments was conducted with three interactions techniques (DT, ITSS and ITOS). The order of the techniques was counterbalanced. Before starting an actual experiment, the users were given an opportunity to run test trials, in order to familiarize themselves with the new interaction techniques and understand the task better. After the trial, it was emphasized to solve the task as fast and as accurate as possible.

4.5 Experiment 1: Tapping

In the first *tapping* experiment we investigated the effect of three described before techniques on the users' performance on both horizontal and vertical screens using different objects' size.

4.5.1 Task

Task - tapping on the circles on the both screens

In this task users were asked to tap on the blue circles with different sizes on both screens. When the touch was performed on the object, user had to hold her finger on it for 0.5 sec till it disappeared. Holding the finger on the object should last till the moment when the new one appears. In this task we measure the time, starting from the moment the next target circle is visible till the moment it is successfully touched by a user. Therefore, the total execution time per one operation included user's reaction time and the time needed to physically move the hand from the previous object to the new target. In order to complete one trial user had to tap on 50 objects, 25 on the vertical screen and 25 on the horizontal. During one trial the size of the object was fixed. The position of the objects was predefined and constant for all the users. We conducted, in summary, 3 experiments for 3 different radii sizes: 63 px (1.5 cm), 126 px (3 cm), 252 px (6 cm). The references between the chosen sizes and real application objects are the following: 63 px refers to the smallest touchable button on a mobile device, for example, Apple iPhone, 126px - to a control element and 252 px - to a document or a picture.

Overall - 450 tapping operations per user

Overall each user had to perform 450 tapping tasks. The experimental design consists of 3 (interaction techniques) $\times$

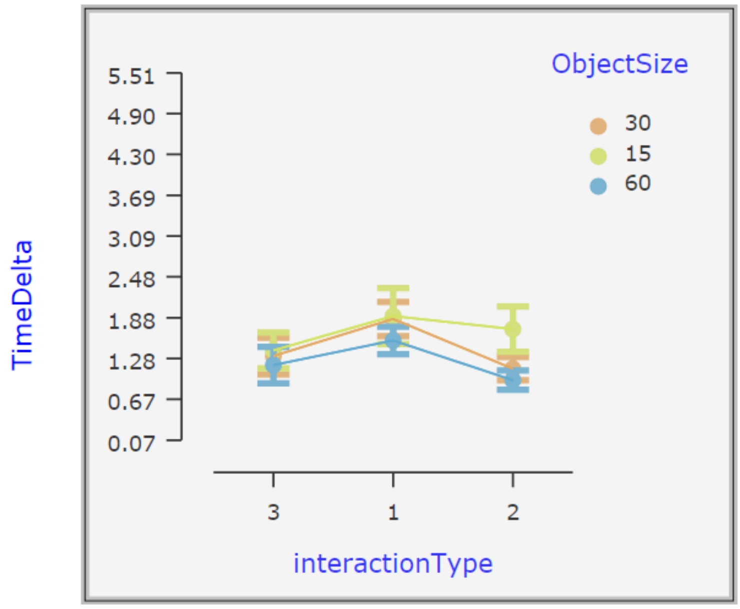

Figure 4.4: 1 - DT, 2 - ITSS, 3 - ITOS.

3 (target sizes) $\times$ 2 (target surfaces) repeated-measure factorial design.

For this task we stated one hypothesis:

H1: The speed of tapping the objects on the vertical screen using ITOS is higher than using ITSS and DT.

4.5.2 Results

The received data was analyzed using repeated measures ANOVA for all independent variables *interaction technique*, *target size* and *target surface*. As far as the data was logarithmically distributed, we transformed the data logarithmically.

ITOS is the fastest for tapping

The ANOVA reported a significant main effect of the factor *interaction technique* $(F(2, 221) = 438.8255; p = 0.0001)$. The Post-hoc Tukey HSD test comparison showed that overall tapping durations using ITOS (mean 0.61 sec) were 32% shorter than while using direct touch (mean 0.9 sec) and

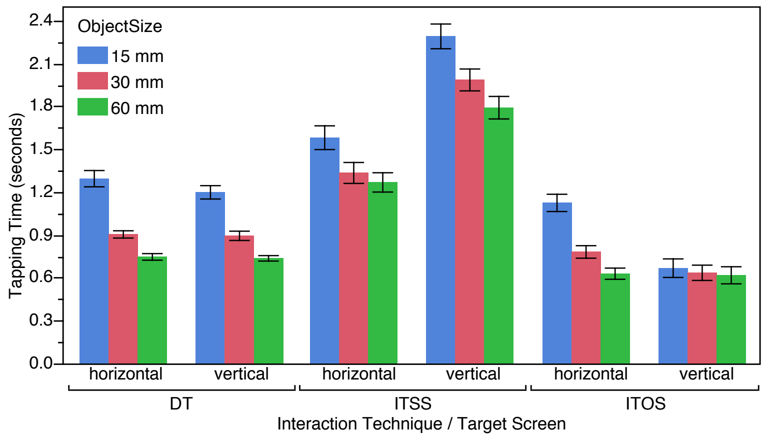

Figure 4.5: Users tapping times using all three interaction techniques in the Tapping experiment. Whiskers denote 95% confidence interval.

60% shorter than ITSS (mean 1.54 sec).

Bigger objects are faster to select

The ANOVA showed a significant main effect of the factor *target size* $(F(2, 221) = 78.7119; p = 0.0001)$. The Post-hoc Tukey HSD comparison showed that the tapping time on the objects with a size of 63px (1.5 cm) (mean 1.14 sec) was 19% longer than on the objects with a size of 126 px (3 cm) (mean 0.92 sec) and 29% longer than on the objects of size 252 px (6 cm) (mean 0.8 sec) for all three techniques. The results for both main effects are shown in Figure4.5. The size of the objects for ITOS was did not have a significant effect. The main effect of the factor *target surface* was not significant.

Tapping on the vertical screen with ITSS is the slowest

The ANOVA showed a significant interaction effect between the factors *interaction technique*, *target size* and *delta time* $(F(4, 221) = 4.301; p = 0.0001)$ (Figure 4.4). The Post-hoc Tukey HSD comparison revealed among other results the following: the tapping time using ITSS on the vertical screen for objects of sizes 63 px (mean time 2.15 sec), 126 px (mean time 1.87 sec) and 252 px (mean time 1.68) was the slowest. The tapping time using ITOS on the vertical screen for objects of sizes 63 px (mean time 0.52 sec), 126 px (mean time 0.5 sec) and 252 px (mean time 0.49 sec) was the fastest, what supports H1.

4.5.3 Discussion

During DT interaction physically traveled distance of the hand is longer

Figure 4.5 illustrates that the ITOS technique was overall the fastest tapping technique in comparison to direct touch and indirect touch screen selection (DT and ITSS). We try to explain the above result by the observation made during the experiment. It tells us more about the way users were interacting with the system and executing the tapping tasks. As it was described in the above section, a new target appears when a user touches the current target circle and holds it for at least 0.5 sec. This leads to the fact that the only action users have to perform are:

1. Find a new target visually.
2. Physically move their hand to the spot on the horizontal or vertical screen, where a new target appeared, and tap.

But for the new target on the vertical screen for ITOS technique user does not have to reach a target physically. As far as an object is highlighted, e.g. it is in the user's focus, she can touch anywhere on the horizontal surface, which takes extremely small amount of time. The task was about tapping on the single object on the surface, so both actions could be executed very fast. Only *lift and tap* gesture is needed on the horizontal surface as long as the object of interest is in the user's focus.

Different sequence of actions among interaction techniques

If we compare ITSS interaction technique with DT and ITOS in the *speed* factor, we see that the needed time for ITSS was comparatively higher than for ITOS or DT. Again the explanation lays in the operation execution process. The interaction on the horizontal surface is the same as for ITOS technique: user has to physically move the arm to reach an object. In order to select an object on the vertical screen she has to run a longer sequence of actions.

1. Look at the vertical screen, touch anywhere (or near the object if the estimation is good enough) on the horizontal screen to make the selecting cursor appear

on the vertical screen, and estimate where the touch is currently mapped to.

2. Move the cursor to the object of interest as using a mouse and perform lift and tap gesture.

As it might be seen from the action sequence, it is a time consuming process. Therefore, tapping task using ITSS interaction technique has the longest execution time. Direct touch still performs good in the tapping tasks, but this might change with an increase of the task duration.

Execution time on horizontal and vertical screens

The interaction time on the horizontal screen for DT and ITOS was comparatively the same, as shown on the Figure 4.5. This is clear, because for both techniques the interaction on the horizontal screen was the same, it is a direct touch. Moreover, the interaction duration on both vertical and horizontal screens for DT is comparatively the same too for all sizes of the objects. This can be explained by the same direct interaction procedure and short task duration. As far as task lasts about 2-3 min, depending on the user, they do not have an opportunity to fully experience the fatigue. We assume, that with the increase of the task duration, these time results for DT on the horizontal screen would differ significantly from the time on the vertical.

ITSS advantages and disadvantages

Obviously, ITSS has its disadvantage in speed, but in the factors of *accuracy*, *convenience* and *reachability* of the distant displays ITSS shows good results. Moreover, detection of the user's focus is easy to do, and the technique is simple to use and understand. As far as we are interested only in the approximate user's focus, in order to estimate the screen we want to interact, we can use low cost head-trackers or cameras. They do not require a high precision and long calibration process, which makes interaction extremely cheap and easy. Additionally, during while using ITSS, users have an opportunity to rest their hands on the horizontal screen and ease an interaction, as known from the common standard computer-based setups.

Shoulder is involved in the direct touch interaction

Using the direct touch condition, a user has to move her entire arm in the mid-air to touch an object on the vertical screen. Not only the hand muscles are involved in the movement, but also the shoulder. Referring to the work by Hammerton and Tickner [1966] this interaction type requires more time than using ITOS.

For ITOS interaction the size of the object does not matter

The next observation which can be implied from the results is that selecting the objects with a bigger size is faster than smaller. This is logical and rather expected for direct touch and ITSS interaction technique. In the case of ITOS interaction technique there was no significant difference in the factor of *object size*. As far as the user has to visually find an object on the vertical screen and tap anywhere on the horizontal, the *size* factor is not highly influential. Because the cursor is mapped to the center of the circle and it does not matter what size a circle has, as long as user is able to recognize and find an object.

Interaction time increases in the situations with multiple objects

In this experiment we looked only at the single object selection task. In the cases, where we have multiple objects concentrated on the same screen close to each other, the object size is going to be influential, and another approach of selecting a needed object from the group has to be taken into account. Also if we increase the number of objects and make the task longer, indirect touch techniques would perform better over time in comparison to direct touch. It might cause the increase in *time* and *efficiency*. Moreover, due to the physiological structure of the human eye, in particular its permanent movement, the selection time would also increase.

4.6 Experiment 2: Parallel Dragging

The purpose of parallel dragging task

After analyzing the users' performance of using three techniques on tapping, we investigated the same set of techniques on *parallel dragging*. *Parallel dragging* means dragging the objects inside the borders of each screen without moving the objects between the screens. While running this experiment we wanted to investigate the influence of the direct touch technique on the users' fatigue for vertical screen interaction, time needed to perform a task on each of the screen and physical distance needed to be traveled on both screens.

4.6.1 Task

To finish the task users had to drag 25 objects on each screen

We asked users to drag blue circles (160 px) to yellow rings (160 px). Each circle-ring pair was displayed within the horizontal and vertical screens. Both pairs were displayed on both screens at the same time and users had to drag the circles on one screen after another. The distance between the ring and circle was the same for all pairs on both screens and equals to 1300 px (30 cm). Users were asked to start each trial with a horizontal screen and continue the next trial on the screen, where they finished the last one. If the position of the circle matches the position of the destination ring within the range of 20 px, the object is accounted as being at the destination. Both objects disappear from the scene, if circle and ring match and user releases the finger. When two circles reached their destination rings, two new circle-ring pairs appear on both screens. In order to complete the overall task users had to drag 25 objects into its target rings on each screen. The measured dependent variables were the receptive dragging times on vertical and horizontal screens. The timer was started at the moment the circle was touched by the user until it was successfully released in its target ring on the same surface. Additionally, we recorded the length of dragging trajectories.

Overall 300 dragging operations were performed

Each user had to perform overall 300 dragging tasks. The experimental design was a 3 (interaction techniques) $\times$ 2 (surface) within group design with repeated measurements. For this experiment we come up with the following hypothesis:

H2: Direct dragging is faster than indirect.

H3: Direct dragging an object on the vertical surface is less accurate than direct dragging on the horizontal and indirect dragging on the vertical surface.

H4: The dragging trajectory length increases over time on the vertical surface using DT.

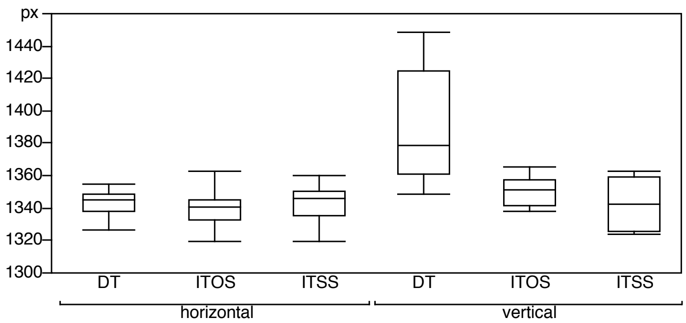

Figure 4.6: User's dragging trajectory length for dragging experiment and all three interaction techniques.

4.6.2 Results

The received data was analyzed using repeated measures ANOVA for all independent variables *interaction technique*, *target size* and *target surface*. As far as the data was logarithmically distributed, we transformed the data logarithmically.

For the variable trajectory length, results of the conducted ANOVA reported a significant main effect of the factors *interaction technique* ($F(2, 65) = 13.4972; p = 0.0001$) and *surface* ($F(2, 65) = 18.5804; p = 0.0001$). The Post-hoc Tukey HSD showed that the dragging trajectory for the DT (mean 1368 px) was significantly longer than the dragging trajectories for ITSS (mean 1347 px) and ITOS (mean 1346 px). It was also shown that the trajectory length on the vertical surface (mean 1362 px) was significant longer than the trajectory length on the horizontal surface (mean 1345 px). The Post-hoc Tukey HSD for the interaction showed that the trajectory length for the DT condition on the vertical surface (mean 1391 px) was significantly longer than the other conditions (mean 1344 - 1350 px), as shown in Figure 4.6).

Fatigue using DT was the largest. Trajectories on the vertical screen are longer than on the horizontal

For the variable time the ANOVA reported a significant main effect of the factors *interaction technique* ($F(2, 65) = 27.6531; p = 0.0001$) and *surface* ($F(2, 65) = 19.2332; p = 0.0001$). The Post-hoc Tukey HSD showed that the dragging time for the DT (mean 1.583 sec) was significantly

DT is the fastest for dragging. Dragging on the horizontal screen takes less time than on the vertical

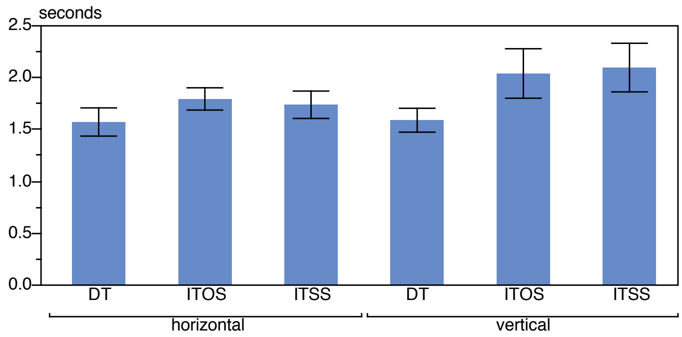

Figure 4.7: Users dragging times using all three interaction techniques subtask in the Dragging experiment. Whiskers denote 95% confidence interval.

shorter than the dragging trajectories for ITSS (mean 1.901 sec) and ITOS (mean 1.8729 sec). It was also shown that the dragging time on the horizontal surface (mean 1.869 sec) was significantly shorter than the dragging time on the vertical surface (mean 1.695 sec). The Post-hoc Tukey HSD for the interaction (Figure 4.7) showed that the dragging time for the ITOS and the ITSS condition on the vertical surface (mean 1.994 sec; 2.08 sec) was significantly longer than the other conditions (mean 1.571, 1.75 sec).

4.6.3 Discussion

Dragging distance increase on the vertical screen for direct touch

Figure 4.6 illustrates the user's dragging trajectory and shows that it is longer for direct dragging an object on the vertical surface in comparison to the horizontal, or indirect dragging on the vertical. The same effect was also observed on the BendDesk (Weiss et al. [2010]). These achieved observations can be explained by the understanding of the users' dragging operation execution and by the loss of accuracy. Indirect touch techniques are slower, but they are more accurate, while for direct touch it is vise verse. Our results can clearly indicate the advantages and disadvantages of the direct and indirect touch techniques. Thus, in the future work an according technique can be chosen according to the needs.

Comparable movement duration for both screens

Dragging duration, however, was comparable for both surfaces for all techniques. This is not surprising, since the operation execution and target's size remained the same. Also in accordance to Fitts' Law (Fitts [1954]), movement duration should stay constant. From the other side, the movement plane did not have any further significant effect on movement duration.

The movement of the arm on the horizontal surface involves most of the times the movement of the forearm and the wrist. As known from the standard computer workspace setups, on the horizontal screen users have an advantage of resting their hands on the horizontal surface, while interacting with the system for dragging operations. Therefore, fatigue is low over time. Long interaction on the vertical screen also brings inaccuracies into the interaction with an increase of fatigue.

Users rest their arms during interaction on the horizontal surface

When users are directly interacting with the vertical surface, the physical movement of the arm involves the upper arm and the shoulder joints. This leads to less accurate interaction, as shown by Hammerton and Tickner [1966], as far as more joints are involved into the interaction process. Our task lasts about 2-3 min, depending on the user, and not all of the users could experience the fatigue during the interaction on the vertical screen. We assume that when it comes to the long daily interaction process on the vertical screen, the effects of fatigue become more significant, and inaccuracy increases.

Upper arm is involved in the vertical screen interaction

As shown in Figure 4.7, the overall fastest technique on the vertical screen in comparison to the other two was direct touch. In comparison to the tapping task, the speed of indirect eye-gaze techniques was slower than for direct touch. This is clearly a disadvantage of the indirect touch techniques over the direct touch, but indirect touch techniques make interaction more comfortable and help with distant display interaction. Moreover, the dragging task is conceptually different and more complicated in the execution in comparison to the tapping.

The fatigue might cause the short time of interaction on the vertical screen

Another influential factor is mapping. It is different for dragging and the presence of additional tracking state, especially for ITSS technique, requires more time. Here, the

Mapping influences the speed

user has to move the cursor to the object, perform lift-and-tap gesture, and then move an object to the ring. With DT user does not waste a lot of time for moving to the circle in the tracking state.

Influence of fatigue

The fatigue, in our opinion, was less influential factor, because, as discussed above, the task did not last long. However, it will change with the change of task duration. As far as users do not want to hold their hands in the mid-air for a long time because of the physical exhaustion, they spend less time for dragging on the vertical screen. Additionally, dragging an object using an indirect touch can be cognitively more challenging in comparison to doing the same action with direct touch.

Long-term vertical screen interaction leads to the loss of accuracy

The distance an object traveled on the vertical screen was longer for direct touch interaction technique than for the other two, what supports H3 and H4. This can be explained by the comfortability of using ITOS and ITSS. As far as users can lean their fingers on the horizontal surface during the indirect touch interaction, the trajectories on the vertical screen stay more or less constant, while for the direct touch interaction the accuracy decrease. Therefore, this resting leads to the less fatigue over a long time of interaction.

Direct touch is exhausting

We support H2, because DT performed better overall on both vertical and horizontal screens, as shown in Figure 4.7. At the end of the experiment most of the users (12) mentioned that dragging the objects while using the direct touch technique was extremely exhaustive. However, the reason for that might lay in the duration of the experiment, which lasts 3-5 minutes, and this time could be too short to show a fatigue effect for direct touch interaction on the vertical screen.

4.7 Experiment 3: Cross Dragging

Purpose of cross dragging - effects of different angles between the target and the object

After exploring and analyzing operations that were involved in the dragging on one surface at the time, we wanted to investigate the interaction technique effect on the user's performance by running the experiment which involves dragging between the screens (*cross dragging*). Moreover, we wanted to observe how different is the interaction on this type of system in comparison to BendDesk and other effects described by Weiss et al. [2010]. In BendDesk system cross dragging experiment was also investigated. They showed that in diagonal dragging operations that involved a horizontal and a vertical surface the user dragging trajectories are significantly longer than in dragging operations that go straight up or downwards.

4.7.1 Task

We conducted the task very similar to the cross dragging by Weiss et al. [2010]. We asked users to drag a blue circle (160 px) displayed on one of the surfaces to a white ring (160 px) placed on the other surface. The task consisted of the following sequence of actions:

1. Drag a blue circle to the edge of the current surface, so a part of it is visible on the second surface.
2. Switch to the other surface (where the second part of the object appeared) and continue dragging a circle on that surface.

The distance between circle-ring pair was constant for all trials - 1631 px (37 cm). There was only one pair of circle and ring in the initial scene. But the angles between ring and circle were not constant and were in the random sequence assigned to $45°$, $30°$, $15°$ to the left, $0°$ (which is straight up or downwards) and $15°$, $30°$, $45°$ to the right (Figure 4.8). Users could start a dragging process either on the horizontal or vertical surface. Starting the dragging on the horizontal surface means the upwards movement and on the vertical - downwards. In order to complete one trial of the task a circle should match the destination ring within a range of 20 px, and user has to release her hand inside. If those actions are successfully performed, both objects disappear from the scene and a new pair of objects appear. Overall, participants had 35 upwards and 35 downwards trials for each of the three interaction techniques. It leads to

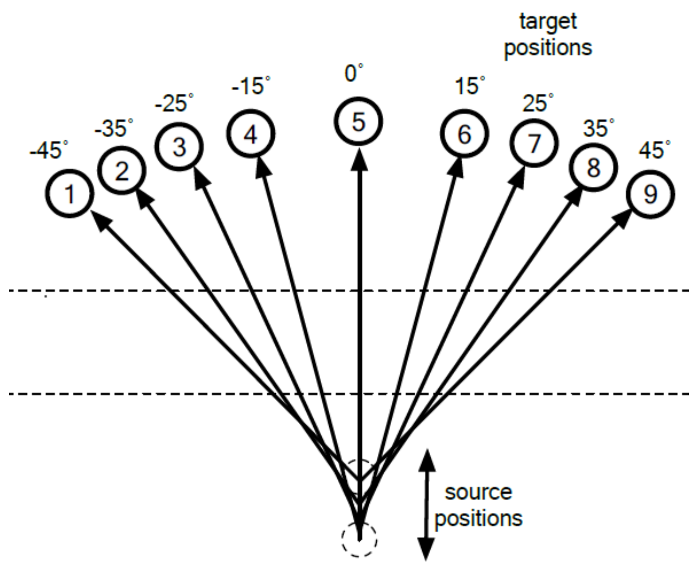

Figure 4.8: Experimental design for cross dragging performance task for upward condition (Weiss et al. [2010]).

210 dragging operations per user for this task. During the time of experiment the system was automatically storing the following data: horizontal and vertical distance, vertical, horizontal and switch time.

The experimental design consists of 3 (interaction techniques) $\times$ 2 (dragging direction) $\times$ 2 (dragging angles) repeated-measure factorial design.

Before the experiment we hypothesized the following outcomes:

H5: Users complete cross dragging operations with ITOS faster than using the other two interaction techniques.

H6: The switching time between the surfaces is the shortest for ITOS interaction technique.

H7: For larger dragging angles the overall dragging trajectory gets longer.

	df		*F*	*p*
Overall time				
Interaction technique	2	276	57.1986	<.0001
Dragging direction	2	276	12.7149	<.0001
Vertical time				
Interaction technique	2	276	10.9089	<.0001
Dragging direction	2	276	146.1459	<.0001
Switching time				
Interaction technique	2	276	33.1058	<.0001
Dragging direction	2	276	15.3929	<.0001
Overall trajectory				
Interaction technique	2	276	65.3526	<.0001
Dragging angle	2	276	29.1385	<.0001
Vertical trajectory length				
Interaction technique	2	276	3.4964	.0316
Dragging angle	2	276	9.5764	<.0001
Dragging direction	2	276	26.3364	<.0001
Horizontal trajectory length				
Interaction technique	2	276	6.7045	.0014
Dragging angle	2	276	9.5764	<.0001
Dragging direction	2	276	12.0905	<.0001

Table 4.1: Significant main effects and interaction for the dependent variables in the CrossDragging experiment.

4.7.2 Results

As far as the data was logarithmically distributed, the following dependent variables were transformed logarithmically: *overall time* (overall task completion time), *vertical time* (time user needed to move an object on the vertical screen), *switching time* (time user needed to switch from the horizontal to the vertical screen and visa versa), *overall trajectory* (the overall physical distance user's finger traveled on both screens), *vertical trajectory length* (the physical distance user's finger traveled on the vertical screen), *horizontal trajectory length* (the physical distance user's finger traveled on the horizontal screen).

Using a repeated measured ANOVA we compared the effect of interaction technique dragging direction, and dragging angle, as well as their interactions on the overall, horizontal and vertical dragging trajectory length, times and switching time. The significant results are shown in Table 4.1. We used the post-hoc Student's t test for the dragging direction variable. The Tukey HSD test was used for the other variables.

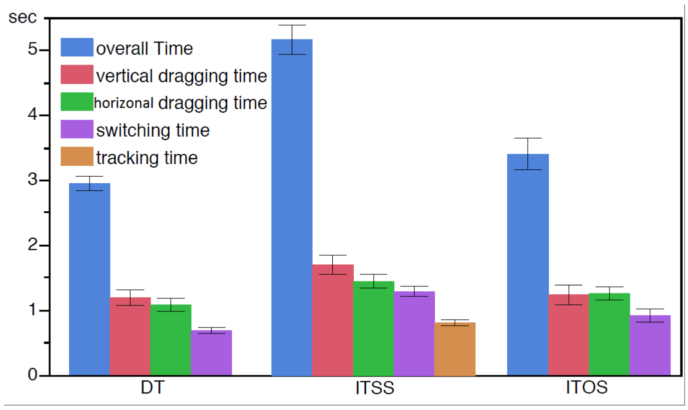

Figure 4.9: Users dragging times for the different subtasks and techniques.

DT is the fastest in cross dragging. Dragging downwards is faster than upwards

The post-hoc test for the *interaction technique* showed that the overall time using ITSS (4.99 sec) was significantly longer than ITOS (3.183 sec) and DT (2.87 sec). Furthermore, upwards dragging (3.87 sec) was significantly slower than the downwards dragging (3.29 sec). The post-hoc test for the vertical time revealed: ITSS (1.54 sec) was significantly smaller than ITOS (1.06 sec) and DT (0.98 sec). Also dragging upwards on the vertical surface (1.78 sec) took significantly longer than dragging downwards (0.77 sec). For the horizontal dragging time the post-hoc test revealed: ITSS (1.29 sec) was significantly slower than ITOS (1.08 sec) and DT (0.95 sec). Dragging upwards (0.94 sec) on the horizontal surface was significantly faster than dragging downwards (1.53 sec).

DT is the fastest in switching between the screens

Switching time using DT (0.59 sec) was significantly faster than ITOS (0.83 sec) and ITSS (1.32 sec). Switching time from the vertical to the horizontal surface (0.74 sec) was shorter than switching from the horizontal to the vertical (1.01 sec).

Overall length is shortest for ITOS

The post-hoc test for the *interaction technique* showed that the *overall length* using ITOS (1568 px) was significantly shorter than for DT (1689 px) and ITSS (1879 px).

Overall and horizontal length gets longer with angle increase

For the *deltaAngle* factor the post-hoc test showed that *overall length* for $0°$ angle (1707 px) was significantly shorter than for $15°$ (1756 px), $30°$ (1809 px) and $45°$ (1851 px).

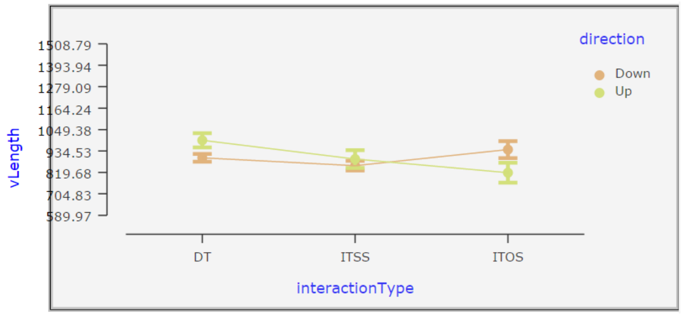

Figure 4.10: Interaction effect between factors of interaction type, direction and trajectory on the vertical screen.

The same tendency was shown for the factor *horizontal length*: for $0°$ angle *horizontal length* (802 px) was significantly shorter than for $15°$ (831 px), $30°$ (857 px) and $45°$ (879 px).

More complicated results were revealed during the analyzes of the interaction effects. The ANOVA showed a significant interaction effect between the factors *interaction technique, direction* and *vertical trajectory length* $(F(2,303) = 15.05; p = 0.0001)$; between the factors *interaction technique, direction* and *horizontal trajectory length* $(F(2,303) = 15.36; p = 0.0001)$ (Figure 4.10).

4.7.3 Discussion

Figure 4.9 illustrates that overall time and time on the vertical surface were longer for ITSS technique than for the other two. This can be explained by referring to the interaction model and its additional *tracking* state. The time needed to move the cursor to the object, user wants to select, increases a lot in opposite to the direct touch interaction, when users could directly and physically reach the target, or look at it while using ITOS technique. Both DT and ITOS require less time to accomplish this task in comparison to ITSS. Moreover, the cross dragging task is more complicated in comparison to the tapping. There was no difference between the users performance using ITOS and DT, therefore H5 was rejected.

Time for ITSS was the longest among techniques

Mid-air arm's distance is an influential factor

The time difference between different techniques for cross dragging might have another influencing factors. For instance, we consider the physical distance the user's arm has to travel in the air during the interaction process, while switching from one surface to another. This distance is the smallest for direct touch interaction, and it is constant in both directions (upwards and downwards). It is the shortest, because it depends only on the physical distance between the vertical and horizontal screens, and equals to the distance between the lower edge of the vertical screen and the upper edge on the horizontal.

The traveled physical hand distance among techniques is not the same

In case of ITOS technique a traveled arm's distance is not constant. One of the dependent factors on the distance is the strategy the user chooses for the interaction process. For example, it depends on where user touches the surface on the horizontal surface, since she is not bounded in the area she can touch, after reaching the border between the screens. Thus, it equals to the distance between the lower edge of the vertical screen and the point on the horizontal surface a user touched. This point lays between the higher and lower edges of the horizontal screen. For ITSS technique the traveling distance is again always constant and equals to the maximum (about the height of the horizontal screen) - the distance between the lower edge of the vertical screen and the lower edge of the horizontal. The same as for direct touch interaction, both upwards and downwards moving direction are equal.

Therefore, the switch time is the longest for ITSS and comparably shorter for ITOS and DT, as shown on Figure 4.9. This leads to the rejection of H6, since in this special case DT outperforms ITOS in switching time.

The result showed that DT was faster than eye-gaze based techniques, which might be surprising. However, this fact can be explained by the following reasons:

1. ITSS has an additional tracking state, which takes more time.

2. DT is more cumbersome and this leads to the faster interaction on the vertical screen. People are not will-

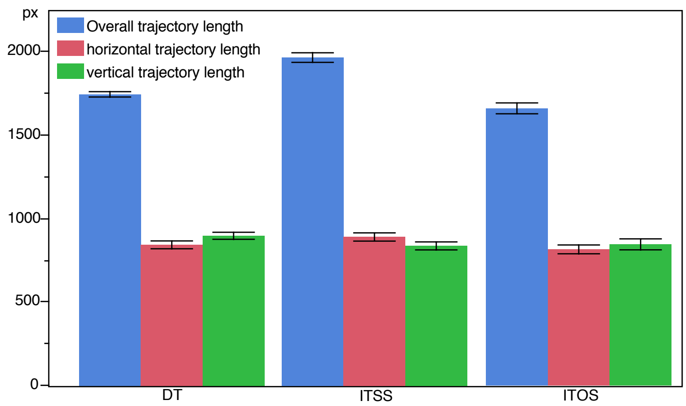

Figure 4.11: Users dragging trajectory length in the Cross-Dragging experiment.

ing to spend much time holding their hands in the air. Thus, this higher interaction speed leads to the shorter time on the vertical screen.

3. Dragging between the screens is more complicated task than tapping.

The difference among techniques in downwards interaction

If we consider the overall time of interaction on the horizontal screen, we can observe that the overall time was the longest for ITSS in comparison to the other two techniques. However, the horizontal time for all three techniques for the upward direction is the same, because of the fact that users repeat the same sequence of actions, what leaves the most influential part for the downwards movement direction. As it was underlined before, the physical movement distance in the air is constant for both upwards and downwards directions using DT technique.

Users could overcome the border without re-grabbing an object for ITOS

In case of ITOS technique users had an opportunity to overcome the border between the screens without re-grabbing an object, but tap on it directly on the horizontal screen and move an object for some time on the horizontal surface, without reaching the destination ring. This could be the reason for the lower time for DT and ITOS techniques in comparison to ITSS, because users did not have to move an arm from the lower edge to the upper edge of the horizontal screen.

Larger angles provoke longer distances

As it was shown by Weiss et al. [2010], the dragging trajectories are longer for larger angles, e.g. more diagonal movement operations cause longer distances. Our results also support this finding. This outcome supports H7 and shows the mentioned above effect not only for curved displays, but also for the systems which combine horizontal and vertical surfaces with a gap between them. Moreover, it is also true for the indirect interaction with a vertical surface. These findings might be explained by the human perception of the distance between the objects. If there is an angle between the objects on the different screens, users do not follow the overall shortest path. First, they drag the object on the one screen, and then on another, but the sum of those paths is bigger than the physical shortest path.

Users minimize the movement on the vertical screen

If we consider the results for the horizontal and vertical dragging trajectories, we can see that trajectories on both surfaces are almost the same. This is surprising, because users can lean their fingers on the horizontal surface and, therefore, rest them during the interaction. It might be explained by the assumption that the duration of our cross dragging task was not long enough, so that users could experience the fatigue. If we increased the duration of the task, the trajectories on the vertical screen could significantly longer than on the horizontal.

4.8 User Study Summary

To sum up the results of our user study, our eye-gaze based techniques are not superior in all aspects over DT, but they have a valuable advantages. Therefore, choosing a technique depends on the situation. User study findings:

1. Tapping on the vertical screen using ITOS is the fastest and ITSS is the slowest method.
2. The size of objects for ITOS does not influence the speed.
3. ITOS and ITSS overcome the *reachability* of the distant display problems.

4. ITOS is more accurate in the interaction process and DT - the least.
5. Trajectories on the vertical screen for parallel dragging are longer than on the horizontal.
6. DT is the fastest for parallel and cross dragging.
7. Dragging on the horizontal screen takes less time than on the vertical.
8. Dragging downwards is faster than upwards.
9. DT is the fastest in switching between the screens.
10. Overall trajectory length is shortest for ITOS.
11. Overall and horizontal trajectory length gets longer with angle increase.

Chapter 5

Application Scenario

In this chapter, we present the description and main concepts of the developed application with a real case scenario, where our proposed techniques can be used.

5.1 General Concepts of the Application

Possible ways of using the evaluated techniques

After the development and the evaluation of the proposed eye-gaze based techniques we decided to show a possible scenario, where such techniques can be used in the real life, and not just stay as a concept which is going to be never used in the future. No doubt, that these techniques can be applied in a variety of possible situations, depending on the needs and a scientist's imagination, starting from the useful scenarios in the service, control or manufacturing fields and ending up with fun applications in different kinds of entertainment.

Our proposed usage scenario - traffic control rooms

In our opinion, one of the possible working environments, where the interaction process could be eased with our eye-gaze based techniques, is the traffic control room. As far as a huge number of information was growing over the last couple of years, the amount of tasks the operators have to handle simultaneously was growing as well. The analysis from Schwarz et al. [2012] showed that operators in the

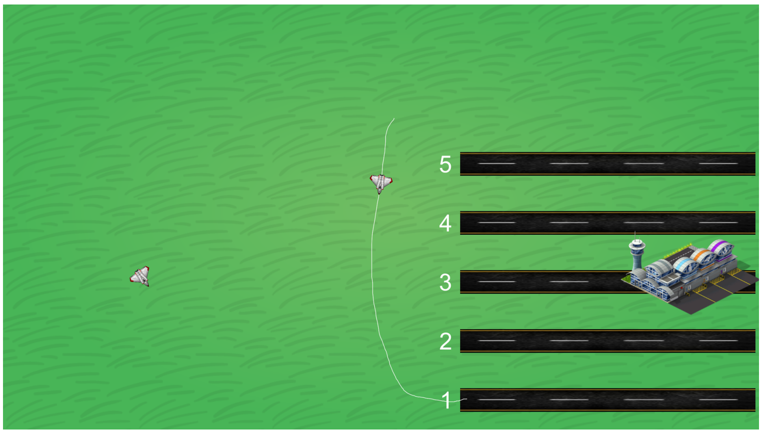

Figure 5.1: Vertical screen: plane is heading to the specified landing line.

traffic control rooms are not adequately supported in their working environments, where they have to monitor, diagnose and manipulate the processes of the current traffic situation. Therefore, we would like to help and assist the control operators in the traffic control situations, by reducing the amount of effort they need to put into the interaction process, and by increasing the interaction speed.

Comparison of ITOS and DT in the chosen scenario

In our user study we revealed a variety of different correlations between the approaches and the gained results, as well as advantages and disadvantages of the proposed and existing techniques. Among the eye-gaze based techniques - ITSS and ITOS - we decided to take ITOS technique, which showed more promising results in comparison to ITSS in the factors of *speed*, *convenience* and *accuracy*. Thus, we would like to show the advantages of ITOS over a traditional direct touch in this application scenario and to demonstrate how the work load of the control room operators might be possibly reduced in the developed application, when it is applied to the real-life scenario.

5.2 Implementation Details

Control of the flights is chosen for the traffic control scenario

As for a traffic case we took a flight control room, where operator has to simultaneously manage the landing of the

Figure 5.2: Horizontal screen: operator sees a zoomed-in picture of the plane, selects and drag'n'drop the transportation.

planes, give them the necessary information, provide a luggage cars, police and buses. In our particular scenario a flight control operators are able to use the described above gaze-based techniques and select planes using an eye-gaze selection.

Flying planes and on-board controls are shown with sprites

In our application we combined a real control-related situation with a game application, so it is a *game-based real life application scenario* of the flight control room. From the one side, our application simulates the flight control surface setup and, from the other, we present all of the flying planes, on-board controls, etc. with two-dimensional image integrated into a scene so-called *sprites* (e.g. game-based representation) as far as we do not have a real controls. The overview information is shown on the vertical screen, e.g. the trajectory of the flying planes, landing lines and a terminal. The information which might overwhelm the vertical screen is shown on the horizontal screen, as well as touch-control buttons.

Flight operator is in charge of two workflows

When the plane is selected, we show an information about the plane on the horizontal surface: flight number, number of passengers on the board, destination, needed transportation supply. As far as on the vertical screen we have a lot of different flying planes, we show a zoomed-in version of the currently selected plane on the horizontal screen and

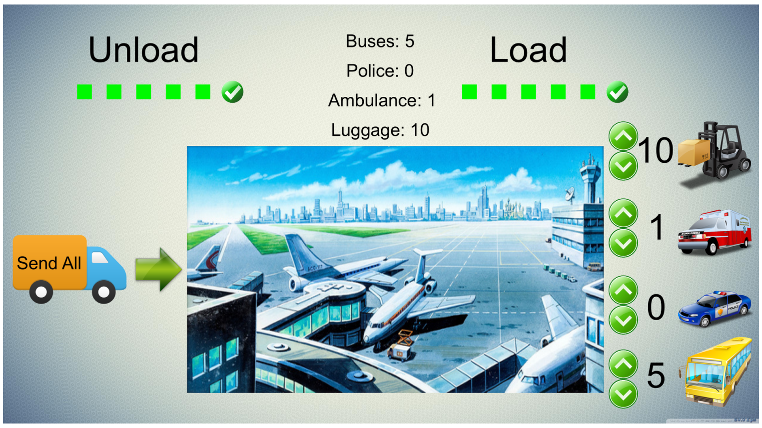

Figure 5.3: Horizontal screen: operator sees the unloading/loading status of the plane and changes the number of transportation if needed.

its real time movement. In the upper part of the horizontal screen an operator can see the number of the transport needed for the selected plane. Thus, she can choose the necessary number of each type of the transport by clicking green arrows up and down, and when the needed number is reached, she needs to drag the transport icon and drop on the plane. If the number of the selected transport matches the number of the needed amount, a green check-box appears next to the needed transportation type (Figure 5.1 and Figure 5.2).

Operator specify a path for a plane using a dragging gesture

Additionally, while selecting a plane an operator can specify a path to follow using a dragging gesture. When a plane is landed, operator selects a terminal on the vertical screen and all landing lines appear on the horizontal screen. After selecting an according line, operator can change a number of the transportation and then with one drag and drop gesture send them to the plane. After performing a drag and drop gesture, a small truck appears on the vertical screen and drives to the plane. When truck reaches the plane an unloading status bar appears on the horizontal screen. Truck drives away when unloading process is done (Figure 5.3).

Drag and drop operation sends a truck to the plane

With the same selection of the transport and sending it by drag and drop gesture, another truck appears. It drives to

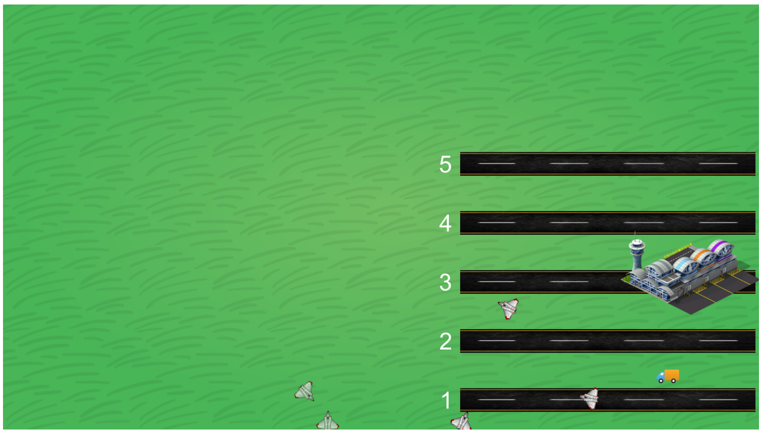

Figure 5.4: After loading a plane it takes off and the truck drives away.

the plane and when it reaches the plane, the loading status bar appears. When the loading process is finished, truck drives back and plane flies away (Figure 5.4). The process is repeated for all the planes presented in the scene. More game screen-shots are presented in the Appendix B.

Therefore, except managing the arrival and departure of the planes, an operator has to manage the control of the transportation process of receiving and delivering the luggage. Thus, an operator is in charge of two flows - planes path control and luggage transport coordination.

Operator is in charge of two transport flows

Considering better results in factors of tapping speed for ITOS technique over ITSS and DT, we decided to use this advantage of the technique for selecting the targets of the vertical screen: flying planes and a terminal. Due to the exhaustion and inconveniences of using dragging on the vertical screen with DT, we keep the dragging functionality on the horizontal screen, by remapping all of the touches which appear on the horizontal screen to the vertical screen in case the user is looking at the vertical screen. Thus, in order to specify a path for a plane an operator has to select a plane with her eye-gaze and provide an according dragging gesture on the horizontal screen. As far as her eye-gaze will be on the vertical screen, all the touches from the horizontal screen are going to be remapped to the vertical one. Therefore, for all of the required operations on

ITOS is used for selection of the targets on the vertical screen. Dragging performed only on the horizontal surface the vertical screen operators would not have to physically reach the screen with their hands. Moreover, they possibly would be able to perform some of the needed operations faster and more accurate.

5.3 Future Extension

We could not evaluate the proposed application, because of the technical problems with eye-gaze tracker. But we find it important to test and compare both DT and ITOS with this application, in order to demonstrate the advantages of ITOS over direct touch interaction.

Multiple problems, which need to be taken into account, might arise during the interaction. For example, the selection of the target of the small size, which is moving comparatively fast on the vertical screen. The calibration and hardware inaccuracies have to be minimized to make an interaction as comfortable and as stable for longer period of time as possible. Selection of the static objects on the vertical with current hardware and calibration procedures was sometimes inaccurate during the user study, which required an additional recalibrations. In the current implementation of the game the planes are moving relatively slow, and the selection should be feasible. However, before bringing this approach to the real life environment, an appropriate evaluation should be done.

One more mapping concern for ITOS is about the length of the trajectories on the horizontal surface. There might be the situations, when the user's finger is too close to the right border of the horizontal surface, so that the user would not be able to finish the dragging within one operation. Thus, as future important addition to the game could be an acceleration of the movement, as it is done on the trackPad or touchPad.

This application can be also extended to the setups with multiple screens. Thus, an operator can have an broader overview of the transport situation in different parts of the airport, as shown on the Figure 5.5. We also assume that

Figure 5.5: Multiple screen working environment (Schwarz et al. [2012]).

with the multiple screen setup ITSS technique will show its advantage of easy selection of the distant screen. If an operator, for example, would need to select only the screen and the content of the distant screen is copied to the horizontal, then ITSS is going to be more feasible technique for this kind of interaction.

Chapter 6

Summary and Future Work

In this last chapter, we summarize the content of the work presented in the thesis and give an outlook of the next steps: where and how ITSS and ITOS can be used in the future.

6.1 Summary and Contributions

In my master's thesis I propose two novel indirect gaze-based interaction techniques, named ITSS (Indirect Touch Screen Selection) and ITOS (Indirect Touch Object Selection), to ease a touch interaction for interactive workspaces.

ITSS

With ITSS the user's gaze determines the target surface and the touch is directed there, using an absolute mapping from the touch surface to the screen. So, if the user looks at the horizontal surface, she can directly interact with objects displayed there, and if the user looks at the vertical surface, her touch is directed there, and she can use an indirect touch to work with objects on the vertical screen. We only use the information where the user is looking at the very moment she creates new touch. As soon as that touch is mapped to one surface, it stays there until the user ends up that touch interaction by lifting her finger again.

ITOS

In ITOS the user does not just use her gaze to determine the right display, instead she selects an object on the screen just by looking at it. Now all the touch events are directly mapped onto this object. If the touch is mapped to one object, it stays there until the user releases her finger from the touch screen.

Tasks summary

Afterwards we evaluated the performance of the suggested techniques and compared them with the traditional direct touch interaction technique on different interaction tasks: *tapping* and *dragging* an object within and between the surfaces. We used a traditional way of interacting on both screens, using direct touch as a baseline. In our tapping experiment users had to constantly switch between tapping objects on horizontal and vertical surface. In our dragging experiments they had to drag objects across the horizontal or vertical surface. And in our cross dragging experiment they had to drag object from horizontal surface to vertical surface and back. Mostly, our indirect touch object selection worked best, although things get more complicated when people are dragging objects between surfaces.

Overall ITOS was about 32% faster than direct touch interaction

The achieved results indicate that ITOS outperforms the direct touch interaction and ITSS interaction technique in terms of tapping speed. Overall, ITOS was about 32% faster than direct touch interaction.

Direct touch has the lowest accuracy

For dragging an object within one screen, direct touch interaction outperformed ITSS and ITOS interaction techniques in terms of speed on the vertical screen. However, within this task the direct touch was the least accurate, and caused the longest touch trajectories on the vertical screen in comparison to the other two techniques.

In cross dragging the movement downwards is faster

In case of cross dragging task, results show that moving an object downwards was faster than upwards over all interaction techniques. The angle between the targets was an influential factor, and by increasing this factor (angle) the physical distance a finger had to travel also increased. Considering the speed, direct touch outperformed ITSS by 42%

and ITOS by 9% interaction techniques.

ITOS is the most preferred techniques

After all, we conclude that ITOS technique in some situations provides an effective and efficient alternative for interaction and manipulation on the interactive workspaces in comparison to traditional direct touch interaction. ITOS technique was also the most preferred among the users.

Bimanual and multifinger interaction for the future

However, in our experiments we only explored the basics and concepts of the interaction by creating tasks which involve single finger interaction with only few objects displayed on the screen. Interactions which involve bimanual and multiple fingers using ITOS for multiple objects in one area on the screen are more difficult and require additional future exploration and evaluation. In order to avoid wrong object selection, the system would require additional identification methods to allow the user to select flawlessly the objects that lay in close proximity to each other.

Application suggestion

We also presented an application suggestion and showed how our gaze-based techniques can be used in the real life scenario. It is a game similar to Flight Control, where an operator has an overview of planes and airport on the vertical screen and an additional information about the flight - on the horizontal. She can select the flying planes and specify the path for them on the vertical screen, and perform all of the additional actions on the horizontal. The evaluation of these technique in the game scenario still has to be done, what is left for the future work.

6.2 Future Work

Our system can be extended to multiple screen setups

In the future both of our proposed interaction techniques, ITSS and ITOS, can be applied in the system setups which combine more than two touch screen surfaces, as shown on the Figure 6.1. In this setup the user has to switch not only between vertical and horizontal screens as for ITSS technique, but switching has to be done for a number of screens presented in the system. Therefore, the user has to activate the vertical screen, which is currently in her focus. The advantages and disadvantages of each technique have to

Figure 6.1: Multitouch and eye-gaze concept for multi-screen environments.

be taken into consideration, so that new approaches could help and ease an interaction process in comparison to the previously existing techniques.

Benefits of ITOS and ITSS in multiscreen environments

Such a multiple screen extension does not provide any screen number limitation, which can be physically integrated in one room. Moreover, eye-gaze based selection can be performed for objects which are displayed on one horizontal and n vertical surfaces, as long as user is able to recognize an object on the surface. But in the user study presented in this thesis users were able to select the objects using eye-gaze based techniques only on the vertical screen, and not on the horizontal. This leads to the limitation that the user cannot interact with objects which are displayed on the vertical screen positioned far away from her. Moreover, multiple horizontal screens might be added to the setup, if needed.

Multi-touch interaction approach

One of the following approach to use a multi-touch interaction using eye-gaze based selection might be the following:

1. User selects an object on the vertical screen using eye-gaze and touches the horizontal screen, thus, the reference is established.
2. Without releasing the finger, user looks at another ob-

ject and performs one more touch. So, another reference (connection) is established.

3. Afterwards she takes control over multiple objects with multiple fingers.

The system is ready for a new object-touch reference establishment as soon as the old one is done. Thus, the user can establish multiple references between the objects on the vertical screen and use a multiple fingers interaction.

Limitation of the object selection on the screen positioned far away from the user

The limitation of the objects selection on the screen positioned far away from the user might be overcome by using ITSS interaction technique. Objects vary in the sizes and selecting them might get tricky, but for big enough screens the selection of the screen in the focus is easier than the selection of an object. Therefore, ITSS might ease the determination whether a user is looking at a large distant screen and allow an interaction on the distance. ITSS interaction can be performed not only by using expensive eye-gaze tracker, but also low-cost head- or eye-trackers. As far as the main idea behind ITSS is to determine an approximate direction of the user's gaze and not the exact position of the eye-gaze on the screen, we can use a head-tracker to grab the current user's focus. Moreover, using the head for selecting a screen is more precise and robust, and can be more comfortable to use, especially for a long time interaction process.

Mobile phones as a future extension of our system

Generally, gaze-based interaction offers a new direction of using the workspaces which integrate one or more horizontal and vertical touch screens. However, the evaluation of such setups with multiple vertical touch screens for the proposed interaction techniques is kept for the future work.

Mobile phones instead of horizontal screen

Another direction can be the exploration of a system which use mobile devices instead one horizontal screen. For example, one could use Apple's iPhone or iPad to interact with multiple vertical distant screens integrated in one interaction environment. Such a mobile extension would lead to new application designs with direct touch as a main interaction method, and gaze as a complementary to it. One of the scenarios can be a public place with a big screen, where users can connect to it using an application and then

Figure 6.2: Sprout HP (sprout.hp.com)

start an interaction, e.g. show a picture to the others.

Limitations and extensions of techniques

Limitations of the proposed techniques in the setup we used for our user study are not yet explored and its possible extension is still questionable, which might be one more point of the future work. A distance acceleration can be an extension to the techniques, in order to avoid the situations of touching the horizontal surface too close to the border, what makes dragging operation impossible in one gesture.

HP product

The newly developed techniques are not just a concept which is never used in the future. HP announced new products called Sprout (Figure 6.2). In order to inform the design of these devices and provide a comfortable interaction, users can use an indirect touch techniques - ITOS and ITSS - for interaction with such systems. The release of such products by HP shows that this area is interesting to dig in and can be explored deeper in the future.

Appendix A

User Study Questionnaire

This appendix contains the questionnaire used in the user study.

Participant ID:

Indirect Touch:

1. Gender: ◯ Male ◯ Female

2. Age: ___________________

3. What is your dominant hand? ◯ Left ◯ Right

4. Was the direct touch (1) interaction technique tiring?

totally disagree		neither		totally agree

5. Was the indirect screen selection (2) interaction technique tiring?

totally disagree		neither		totally agree

6. Was the indirect object selection (3) interaction technique tiring?

totally disagree		neither		totally agree

6. Which interaction technique do you prefer for tapping?

◯ Direct touch ◯ Indirect screen selection ◯ Indirect object selection

7. Which interaction technique do you prefer for single screen dragging?

◯ Direct touch ◯ Indirect screen selection ◯ Indirect object selection

Figure A.1: User Study Questionnaire page1.

Participant ID:

8. Which interaction technique do you prefer for cross screen dragging?

◯ Direct touch ◯ Indirect screen selection ◯ Indirect object selection

9. Which interaction technique allowed you to select the object the fastest?

◯ Direct touch ◯ Indirect screen selection ◯ Indirect object selection

Any further comments?

Figure A.2: User Study Questionnaire page2.

Appendix B

Application screenshots

This appendix contains an overview and an additional screen shots of the application described in the Chapter 5 **"Application Scenario"**.

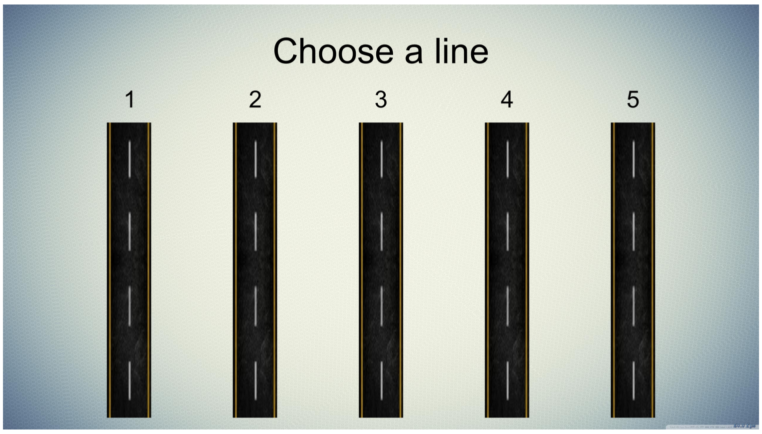

Figure B.1: When an operator clicks on the terminal on the vertical screen 5 landing lines appear on the horizontal screen.

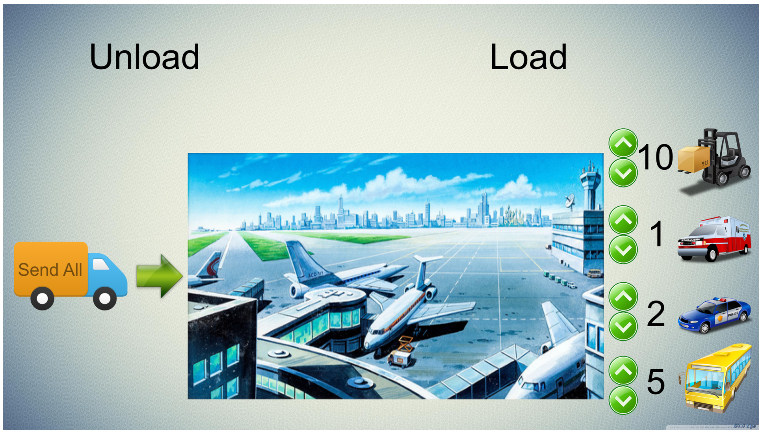

Figure B.2: After selecting a landing line an operator can change the number of the transportation.

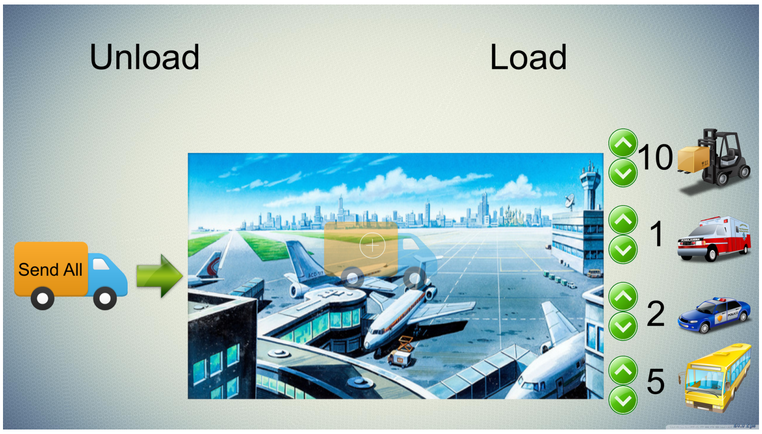

Figure B.3: In order to send the transportation to the landed plane an operator has to drag a truck on the right side and drop it on the plane in the middle.

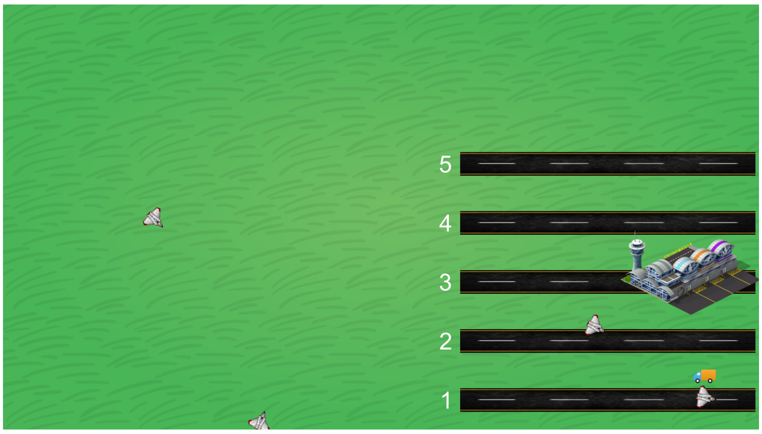

Figure B.4: When a truck was dropped, a smaller version of it appears on the vertical screen and drives to the plane.

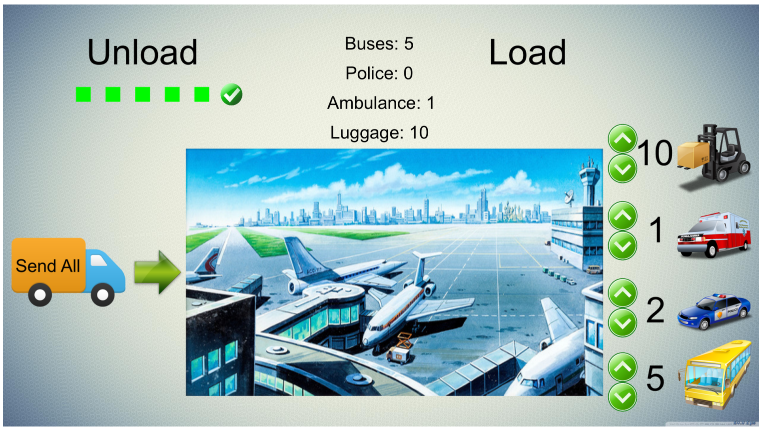

Figure B.5: When the *unloading* process is done, the information about the *loading* transportation appears on the screen and can be started with the same drag and drop operation.

Bibliography

Mark Ashdown, Kenji Oka, and Yoichi Sato. Combining Head Tracking and Mouse Input for a GUI on Multiple Monitors. In *CHI EA '05*, pages 1188–1191. ACM, 2005. ISBN 1-59593-002-7. doi: 10.1145/1056808.1056873. URL `http://doi.acm.org/10.1145/1056808.1056873`.

Xiaojun Bi, Tovi Grossman, Justin Matejka, and George Fitzmaurice. Magic desk: Bringing multi-touch surfaces into desktop work. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2511–2520, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1979309. URL `http://doi.acm.org/10.1145/1978942.1979309`.

Jurek Breuninger, Christian Lange, and Klaus Bengler. Implementing gaze control for peripheral devices. In *Proceedings of the 1st International Workshop on Pervasive Eye Tracking & Mobile Eye-based Interaction*, PETMEI '11, pages 3–8, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0930-1. doi: 10.1145/2029956.2029960. URL `http://doi.acm.org/10.1145/2029956.2029960`.

William Buxton. A three-state model of graphical input. In *Proceedings of the IFIP TC13 Third Interational Conference on Human-Computer Interaction*, INTERACT '90, pages 449–456, Amsterdam, The Netherlands, The Netherlands, 1990. North-Holland Publishing Co. ISBN 0-444-88817-9. URL `http://dl.acm.org/citation.cfm?id=647402.725582`.

Francois Coldefy and Stphane Louis dit Picard. Digitable: an interactive multiuser table for collocated and remote collaboration enabling remote gesture visualization. *2013*

IEEE Conference on Computer Vision and Pattern Recognition, 0:1–8, 2007. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.2007.383471.

Katherine Everitt, Chia Shen, Kathy Ryall, and Clifton Forlines. Multispace: Enabling electronic document micro-mobility in table-centric, multi-device environments. In *Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, TABLETOP '06, pages 27–34, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2494-X. doi: 10.1109/TABLETOP.2006.23. URL `http://dx.doi.org/10.1109/TABLETOP.2006.23`.

Paul M Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954.

David Fono and Roel Vertegaal. EyeWindows: Evaluation of Eye-controlled Zooming Windows for Focus Selection. In *Proc. CHI '05*, pages 151–160. ACM, 2005. ISBN 1-58113-998-5. doi: 10.1145/1054972.1054994. URL `http://doi.acm.org/10.1145/1054972.1054994`.

Clifton Forlines, Daniel Wigdor, Chia Shen, and Ravin Balakrishnan. Direct-touch vs. mouse input for tabletop displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 647–656, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9. doi: 10.1145/1240624.1240726. URL `http://doi.acm.org/10.1145/1240624.1240726`.

M. Hammerton and A. H. Tickner. An Investigation into the Comparative Suitability of Forearm, Hand and Thumb Controls in Acquisition Tasks. *Ergonomics*, 9:125–130, 1966.

Robert J. K. Jacob. What you look at is what you get: Eye movement-based interaction techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 11–18, New York, NY, USA, 1990. ACM. ISBN 0-201-50932-6. doi: 10.1145/97243.97246. URL `http://doi.acm.org/10.1145/97243.97246`.

Jakob Leitner, James Powell, Peter Brandl, Thomas Seifried, Michael Haller, Bernard Dorray, and Paul To. Flux: A tilting multi-touch and pen based surface. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 3211–3216, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-247-4. doi: 10.1145/1520340.1520459. URL `http://doi.acm.org/10.1145/1520340.1520459`.

Diako Mardanbegi and Dan Witzner Hansen. Mobile gaze-based screen interaction in 3d environments. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*, NGCA '11, pages 2:1–2:4, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0680-5. doi: 10.1145/1983302.1983304. URL `http://doi.acm.org/10.1145/1983302.1983304`.

Meredith Ringel Morris, A. J. Bernheim Brush, and Brian R. Meyers. Reading revisited: Evaluating the usability of digital display surfaces for active reading tasks. In *In Proc. of TABLETOP 2007*, pages 79–86, 2007.

Merrie Morris, A.J. Brush, and Brian Meyers. A field study of knowledge workers' use of interactive horizontal displays. In *Proceedings of IEEE Tabletops and Interactive Surfaces 2008*. IEEE Computer Society, October 2008. URL `http://research.microsoft.com/apps/pubs/default.aspx?id=79359`.

Christian Müller-Tomfelde, Anja Wessels, and Claudia Schremmer. Tilted tabletops: In between horizontal and vertical workspaces. In *Third IEEE International Workshop on Tabletops and Interactive Surfaces (Tabletop 2008), October 1-3 2008, Amsterdam, The Netherlands*, pages 49–56, 2008. doi: 10.1109/TABLETOP.2008.4660183. URL `http://dx.doi.org/10.1109/TABLETOP.2008.4660183`.

Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P. Irani, and Michel Beaudouin-Lafon. High-precision Pointing on Large Wall Displays Using Small Handheld Devices. In *Proc. CHI '13*, pages 831–840. ACM, 2013. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470773. URL `http://doi.acm.org/10.1145/2470654.2470773`.

Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. Gaze-touch: Combining gaze with multi-touch for interaction on the same surface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 509–518, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. doi: 10.1145/2642918.2647397. URL http://doi.acm.org/10.1145/2642918.2647397.

Jun Rekimoto and Masanori Saitoh. Augmented surfaces: A spatially continuous work space for hybrid computing environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 378–385, New York, NY, USA, 1999. ACM. ISBN 0-201-48559-1. doi: 10.1145/302979.303113. URL http://doi.acm.org/10.1145/302979.303113.

Dominik Schmidt, Florian Block, and Hans Gellersen. A Comparison of Direct and Indirect Multi-touch Input for Large Surfaces. In *Proc.INTERACT '09*, pages 582–594, 2009. URL http://www.springerlink.com/content/c1689n68104100p1/.

Tobias Schwarz, Simon Butscher, Jens Mueller, and Harald Reiterer. Content-aware navigation for large displays in context of traffic control rooms. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, pages 249–252, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1287-5. doi: 10.1145/2254556.2254601. URL http://doi.acm.org/10.1145/2254556.2254601.

Jeffrey S. Shell, Roel Vertegaal, Daniel Cheng, Alexander W. Skaburskis, Changuk Sohn, A. James Stewart, Omar Aoudeh, and Connor Dickie. Ecsglasses and eyepliances: Using attention to open sociable windows of interaction. In *Proceedings of the 2004 Symposium on Eye Tracking Research & Applications*, ETRA '04, pages 93–100, New York, NY, USA, 2004. ACM. ISBN 1-58113-825-3. doi: 10.1145/968363.968384. URL http://doi.acm.org/10.1145/968363.968384.

J. David Smith and T. C. Nicholas Graham. Use of eye movements for video game control. In *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in*

Computer Entertainment Technology, ACE '06, New York, NY, USA, 2006. ACM. ISBN 1-59593-380-8. doi: 10.1145/1178823.1178847. URL `http://doi.acm.org/10.1145/1178823.1178847`.

Sophie Stellmach and Raimund Dachselt. Look & touch: Gaze-supported target acquisition. In *Proc. CHI '12*, CHI '12, pages 2981–2990. ACM, 2012. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2208709. URL `http://doi.acm.org/10.1145/2207676.2208709`.

Sophie Stellmach and Raimund Dachselt. Still Looking: Investigating Seamless Gaze-supported Selection, Positioning, and Manipulation of Distant Targets. In *Proc. CHI '13*, pages 285–294. ACM, 2013. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470695. URL `http://doi.acm.org/10.1145/2470654.2470695`.

Sophie Stellmach, Sebastian Stober, Andreas Nürnberger, and Raimund Dachselt. Designing Gaze-supported Multimodal Interactions for the Exploration of Large Image Collections. In *Proc. NGCA '11*, pages 1:1–1:8. ACM, 2011. ISBN 978-1-4503-0680-5. doi: 10.1145/1983302.1983303. URL `http://doi.acm.org/10.1145/1983302.1983303`.

Bruce Tognazzini. The “starfire” video prototype project: A case history. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pages 99–105, New York, NY, USA, 1994. ACM. ISBN 0-89791-650-6. doi: 10.1145/191666.191712. URL `http://doi.acm.org/10.1145/191666.191712`.

Jayson Turner. Cross-device Eye-based Interaction. In *Proc. UIST AP '13*, pages 37–40. ACM, 2013. ISBN 978-1-4503-2406-9. doi: 10.1145/2508468.2508471. URL `http://doi.acm.org/10.1145/2508468.2508471`.

Jayson Turner, Andreas Bulling, and Hans Gellersen. Combining Gaze with Manual Interaction to Extend Physical Reach. In *Proc. PETMEI '11*, pages 33–36, 2011. ISBN 978-1-4503-0930-1. doi: 10.1145/2029956.2029966. URL `http://doi.acm.org/10.1145/2029956.2029966`.

Jayson Turner, Andreas Bulling, Jason Alexander, and Hans Gellersen. Cross-device Gaze-supported Point-to-point Content Transfer. In *Proc. ETRA '14*, pages 19–26. ACM, 2014. ISBN 978-1-4503-2751-0. doi: 10.1145/2578153.2578155. URL `http://doi.acm.org/10.1145/2578153.2578155`.

Simon Voelker, Christine Sutter, Lei Wang, and Jan Borchers. Understanding Flicking on Curved Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 189–198, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2207703. URL `http://doi.acm.org/10.1145/2207676.2207703`.

Simon Voelker, Chat Wacharamanotham, and Jan Borchers. An evaluation of state switching methods for indirect touch systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 745–754, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470759. URL `http://doi.acm.org/10.1145/2470654.2470759`.

Malte Weiss, Simon Voelker, Christine Sutter, and Jan Borchers. BendDesk: Dragging Across the Curve. In *Proc. ITS '10*, pages 1–10, 2010. ISBN 978-1-4503-0399-6. doi: 10.1145/1936652.1936654. URL `http://doi.acm.org/10.1145/1936652.1936654`.

Daniel Wigdor, Chia Shen, Clifton Forlines, and Ravin Balakrishnan. Effects of display position and control space orientation on user preference and performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 309–318, New York, NY, USA, 2006. ACM. ISBN 1-59593-372-7. doi: 10.1145/1124772.1124819. URL `http://doi.acm.org/10.1145/1124772.1124819`.

Daniel Wigdor, Gerald Penn, Kathy Ryall, Alan Esenther, and Chia Shen. Living with a tabletop: Analysis and observations of long term office use of a multi-touch table. In *Tabletop'07*, pages 60–67, 2007.

Raphael Wimmer, Fabian Hennecke, Florian Schulz, Sebastian Boring, Andreas Butz, and Heinrich Hussmann.

Curve: Revisiting the Digital Desk. In *Proc. NordiCHI '10*, pages 561–570. ACM, 2010. ISBN 978-1-60558-934-3. doi: 10.1145/1868914.1868977. URL `http://doi.acm.org/10.1145/1868914.1868977`.

Shumin Zhai, Carlos Morimoto, and Steven Ihde. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proc. CHI '99*, pages 246–253. ACM, 1999. ISBN 0-201-48559-1. doi: 10.1145/302979.303053. URL `http://doi.acm.org/10.1145/302979.303053`.

Index

Typeset December 4, 2014